digital

MAINDEC CHANGE NOTICE

254 MCN

DG
P.O. Box 4313
Tucson, Arizona 85717

CHANGE NO. 11-DZDLA-A- 1

Sheet 1 of 1

AUTHOR	PROGRAM DATE	PRODUCT LINE	MAINDEC NUMBER
J. Friedrich DATE EXT. 3/9/73 3142	JUNE 72	PDP-11 V68 06555	11-DZDLA-A

PROGRAM NAME DL11 Off Line List **DEVICE** DL11C,D,E

ITEM

1. This patch is for fast 11 processors with DL11's with slow EIA level converters. The patch is just a delay mechanism.

```
φφ7542:φφ4737          ;JSR PC,STALL
φφ7544:φ17000          ;
φφ7546;φφφ240          ;NOP

φ17000:φ17737          ;MOV @RXCSR,@#RXCSRT
       162400
       φφ1602
       φφφ207          ;RTS PC
```

The stall time is the time it takes to execute a JSR PC, STALL.

old

```
7542: φ17767
      171636
      172φ32
      φ32767
```

old

```
17φφφ:
```

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DEDLA-A-D

PRODUCT NAME: DL11-E,C/D OFF LINE TEST

DATE CREATED: JUNE 1, 1972

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: ROBERT WHITTON

COPYRIGHT © 1972
DIGITAL EQUIPMENT CORPORATION

MAIN DEC CHANGE NOTICE
MAY BE REQUIRED FOR
PROGRAM TO OPERATE

DIGITGRAPH
P. O. Box 4240
Tucson, Arizona

1, ABSTRACT

TWO SEPARATE DIAGNOSTIC PROGRAMS ARE PROVIDED FOR THE DL11-E (ASYNCHRONOUS MODEM INTERFACE), MAINDEC-11-DZDLA-A (DL11-E OFF LINE TESTS) AND MAINDEC-11-DZDLB-A (DL11-E ON LINE TESTS). THE OFF LINE TEST TESTS ALL DL11-E LOGIC. THE OFF LINE TESTS DO NOT REQUIRE THE USE OF A MODEM, HOWEVER A SPECIAL JUMPER CONNECTOR H315 IS REQUIRED. THE ON LINE TESTS ARE ESSENTIALLY DATA RELIABILITY TESTS REQUIRING THE USE OF MODEMS AND A SUITABLE TERMINAL DEVICE.

THE DL11-C AND DL11-D CAN ALSO BE TESTED WITH THIS OFF LINE TEST. THESE ARE BOTH TESTED IN MAINTENANCE MODE AND ONLY THOSE TESTS MARKED C,D IN THE TEST NUMBER ARE EXECUTED. IN ORDER TO TEST C AND D VERSIONS IT IS NECESSARY TO MODIFY THE TABLE AT LOCATION 1300 ACCORDING TO THE INSTRUCTIONS CONTAINED THERE.

TESTS WHICH ARE NOT EXECUTED FOR DL11C-D CAN BE PERFORMED BY USING THE SELECT SWITCH OPTION (SR9). TEST 56 IS A DATA TEST WHICH CAN BE USED FOR CABLE TESTING DL11-D'S. WARNING--A FAILURE IN THIS TEST MAY OCCUR DUE TO A SPLIT BAUD RATE OF RCVTR/TXVTR.

THIS DOCUMENT DESCRIBES THE OFF LINE TESTS.

THE AVAILABLE TESTS ARE:

PRG0	INPUT/OUTPUT LOGIC TESTS
PRG1	TRANSMITTER SCOPE LOOP
PRG2	RECEIVER SCOPE LOOP
PRG3	SINGLE CHARACTER MAINT. MODE DATA TEST
PRG4	SPECIAL BINARY COUNT MAINTENANCE MODE DATA TEST

2, REQUIREMENTS

2,1 EQUIPMENT

A, PDP 11 SYSTEM

B, DL11-E OR DL11-C OR DL11-D

C, SPECIAL JUMPER CONNECTOR H315 (SEE DL11 MAINTENANCE MANUAL FOR DETAILED DESCRIPTION) IF DL11-E.

2,2 STORAGE

THIS PROGRAM USES ALL OF CORE (4K) EXCEPT THAT AREA RESERVED FOR THE BOOTSTRAP AND ABSOLUTE LOADERS.

3, LOADING PROCEDURE

THE ABSOLUTE LOADER IS USED TO LOAD THE PROGRAM.

4, USE PROCEDURE
BEFORE STARTING ANY OF THE SELECTABLE PROGRAMS MAKE SURE
THAT THE TTY IS IN REMOTE MODE, AND THAT THE PROGRAM SELECTED
IS A LEGAL PROGRAM, IE; SR 0-2-0-4, OTHERWISE AN ERROR MESSAGE
WILL OCCUR;

A MAP OF DEVICES PRESENT WILL BE TYPED AT RUN TIME; THIS MAP WILL
NOT BE TYPED OUT AGAIN UNLESS THE PROGRAM IS RELOADED OR IS
RESTARTED AT LOCATION 210, A RESTART FROM THIS LOCATION WILL CAUSE
THE MAP OF DEVICES TO BE TYPED OUT AGAIN AND THEN A NORMAL START
WILL OCCUR;

4.1 PRG0 INPUT/OUTPUT LOGIC TESTS

A, LOAD ADDRESS = 000200 (RESTART LOAD ADDR, = 000204)
LOAD SR 0-2 = 0, AND PRESS START SWITCH;
THE DIAGNOSTIC WILL IDENTIFY THE PROGAM YOU SELECTED,
DISCONNECT THE DL11-E FROM THE MODEM AND INSERT THE JUMPER CON-
NECTOR IN THE MODEM END OF THE CABLE, AND PRESS CONTINUE,
NOTE, IF THE CABLE IS LEFT CONNECTED TO THE MODEM THE FOL-
LOWING TESTS WILL FAIL!
AT22, AT23, AT25, AT30, AT32, AT50

B, THE PROGRAM WILL TYPE OUT INSTRUCTIONS TO SET IN THE DESIRED
SR OPTIONS; PRESS CONTINUE WHEN THE OPTIONS ARE IN THE SR;
THE AVAILABLE OPTIONS ARE!
SR 0-6 ROUTINE TO BE RUN (IF ENABLED BY SR9)
SR7 DISABLE STALL MODE
SR9 LOOP SELECTED ROUTINE
SR10 HALT AT END OF CURRENT TEST
SR11 INHIBIT ITERATION
SR12 SELECT LINE NUMBER AND LOCK ON IT
SR13 INHIBIT PRINTOUT
SR14 SCOPE
SR15 HALT ON ERROR;

C; THE PROGRAM WILL NOW REQUEST THE LINE #(IF SR12=1) YOU WISH TO
TEST, LOAD THE LINE # AS REQUESTED AND PRESS CONTINUE;
LINE NUMBER REFERS TO THE ADDRESSES TO WHICH THE DL11-E RESPONDS;

LINE 00 77560X	LINE 10 77570X	LINE 20 77600X	LINE 30 77610X
LINE 01 77562X	LINE 11 77572X	LINE 21 77602X	LINE 31 77612X
LINE 02 77564X	LINE 12 77574X	LINE 22 77604X	LINE 32 77614X
LINE 03 77566X	LINE 13 77576X	LINE 23 77606X	LINE 33 77616X
LINE 04 77550X	LINE 14 77540X	LINE 24 77610X	LINE 34 77620X
LINE 05 77552X	LINE 15 77542X	LINE 25 77612X	LINE 35 77622X
LINE 06 77554X	LINE 16 77544X	LINE 26 77614X	LINE 36 77624X
LINE 07 77556X	LINE 17 77546X	LINE 27 77616X	

D, THE PROGRAM WILL NOW BEGIN TESTING THE DL11-E OR C/D YOU SELECTED, ALL DL11'S WILL BE TESTED AUTOMATICALLY AND SEQUENTIALLY UNLESS SR12 IS SELECTED;
NOTE! ALL LOGIC TESTS WILL NOT BE RUN AUTOMATICALLY, THERE ARE TWO TESTS WHICH REQUIRE MANUAL INTERVENTION WHICH ARE USED TO TEST THE SPEED SELECTION SWITCHES, THESE ARE TESTS T34,T40, TO EXECUTE THESE TESTS USE SR0 AND SR 0-6 TO SELECT THEM',

E, REFER TO SECTION 5',1',2 FOR ERROR DESCRIPTION

F, AFTER ONE COMPLETE PASS THE BELL WILL RING FOLLOWED BY "END PASS # " WITH THE NUMBER OF PASSES COMPLETED SINCE PROGRAM LAST STARTED AND THE DEVICE ADDRESS UNDER TEST AND ITS TRAP VECTOR',

4',2 PRG1 - TRANSMITTER SCOPE LOOP

A, LOAD ADDRESS = 000200 (RESTART = 000204)
LOAD SR 3-2 = 1, AND PRESS START SWITCH',
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND REQUEST THE LINE # YOU WISH TO TEST, LOAD THE LINE # AS REQUESTED AND PRESS CONTINUE',

B, THE PROGRAM WILL REQUEST A CHARACTER CODE, AND A DELAY TIME, THE CHARACTER CODE IS THE DATA THE DL11-E WILL TRANSMIT AND THE DELAY IS THE TIME ELAPSED BETWEEN SUCCESSIVE TRANSMISSIONS OF ONE CHARACTER', PRESS CONTINUE WHEN THIS IS DONE,

C', THE PROGAM WILL RUN WITHOUT ERROR OR END TYPEOUTS',

4',3 PRG2 - RECEIVER SCOPE LOOP

A, LOAD ADDRESS = 000200 (RESTART = 000204)
LOAD SR 3-2 = 2, AND PRESS START,
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND REQUEST THE LINE # YOU WISH TO TEST, LOAD THE LINE # AS REQUESTED AND PRESS CONTINUE',

B, THE PROGRAM WILL REQUEST A TEST CHARACTER CODE, AND A DELAY TIME, THE CHARACTER CODE IS THE DATA THAT THE DL11-E WILL BE TRANSMITTING AND THE DELAY IS THE ELAPSED TIME BETWEEN SUCCESSIVE CHARACTERS, PRESS CONTINUE WHEN THIS IS DONE,

C, THE PROGRAM WILL NOW RUN WITHOUT ERROR OR END TYPEOUTS',

4,4 PRG3 - SINGLE CHARACTER MAINT MODE DATA TEST

A, LOAD ADDRESS = 000200 (RESTART = 000204)
LOAD SR 0-2 = 3, AND PRESS START,
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND REQUEST THE LINE # YOU WISH TO TEST, LOAD THE LINE # AS REQ-UESTED AND PRESS CONTINUE',

B', THE PROGRAM WILL REQUEST A TEST CHARACTER', LOAD THE TEST CHAR-ACTER AND PRESS CONTINUE',

C', THE PROGRAM WILL NOW RUN CONTINUOSLY REPORTING ANY DATA FAIL-URES,

4,5 PRG4 - SPECIAL BINARY COUNT MAINT', MODE DATA TEST

A, LOAD ADDRESS = 000200
LOAD SR 0-2 = 4, AND PRESS START,
THE DIAGNOSTIC WILL IDENTIFY THE PROGRAM YOU SELECTED, AND REQUEST THE LINE # YOU WISH TO TEST, LOAD THE LINE # AS REQ-UESTED AND PRESS CONTINUE',

B', THE PROGRAM WILL BEGIN TESTING THE LINE YOU SELECTED,
AND REPORT ANY DATA ERRORS',

5, PROGRAM DESCRIPTIONS

5,1 PRG0 - INPUT/OUTPUT LOGIC TESTS

THE INPUT/OUTPUT LOGIC TESTS CONSIST OF 57(8) ROUTINES WHICH MAY BE RUN IN SEQUENTIAL ORDER OR INDIVIDUALLY LOOPED (SEE SECT 4,1, C FOR SWITCH SETTINGS), THE JUMPER CONNECTOR MUST BE INSERTED BEFORE STARTING IF DL11-E,

5.1.1 ROUTINE DESCRIPTIONS

ROUTINE	TESTS
AT0-AT3	ADDRESSABILITY OF CSRS & DBRS
AT4-AT27	DIDDLES ALL BITS IN THE CSRS AND CHECKS THAT THEY CAN BE READ/WRITTEN PROPERLY.
AT31-AT32	PROPER OPERATION OF RESET INSTRUCTION
AT33	PROPER OPERATION OF READY BIT
AT34	PROPER OPERATION OF TRANSMIT SPEED SELECTION
AT35-AT37	PROPER OPERATION OF DONE BIT
AT40	PROPER OPERATION RECEIVER SPEED SELECT
AT41	PROPER OPERATION OF DATA OVERRUN
AT42-AT52	PROPER OPERATION OF INTERRUPTS
AT53	READING RXCSR DOES NOT CLEAR DONE
AT54	ERROR CAUSES INTERRUPT
AT55	DATA TEST MAINTENANCE MODE
AT56	DATA TEST WITH JUMPER
AT57	PROPER OPERATION OF BREAK BIT

5.1.2 ERROR DESCRIPTION

IF A ROUTINE FAILS AND THE INHIBIT PRINTOUT SWITCH IS NOT ENABLED ($SR13) A PRINTOUT RESULTS; THE PRINTOUT FORMAT IS:

T(ROUTINE#) PC=(PC OF ERROR CALL) RXCSR=(ADDRESS OF DEVICE UNDER TEST) AND AN ADDITIONAL/MESSAGE (IF APPLICABLE)

T005 PC=XXXX RXCSR=XXXX

T122 PC=XXXX RXCSR=XXXX DATA S/B=XX-WAS=-XX
INDICATING A DATA ERROR

TO RESUME TESTING PRESS CONTINUE.
IF THE VECTOR PROVIDED BY THE INTERRUPTING DL11-E IS INCORRECT A TRAP TO THE WRONG LOCATION WILL OCCUR AND AN ERROR MESSAGE WILL OCCUR.

5.1.3 JUMPER CONNECTOR

THE JUMPER CONNECTOR TESTS THOSE F/F'S, GATES (RING INDICATOR, CARRIER TRANSITION, CLEAR TO SEND, AND SUPERVISORY RECEIVE DATA) WHICH CANNOT BE TESTED UNLESS A DATA SET IS ACTUALLY CONNECTED TO THE DL11-E. IN ADDITION TO TESTING DL11-E LOGIC THE JUMPER ALSO TESTS CABLE WIRING TO/FROM THE DL11-E/DATA SET. THE FOLLOWING TESTS WILL FAIL IF THE CABLE IS NOT INSALLED IN THE DL11-E:

AT22,AT23,AT25,AT3#,AT32,AT36

5,2 PRG1-TRANSMITTER SCOPE LOOP

THE PURPOSE OF PRG1 IS TO ALLOW SCOPING OF TRANSMITTER FUNCTIONS IN A RUN CONDITION USING USER SPECIFIED DL11-E PARAMETERS AND DATA. NO ERROR PRINTOUTS ARE PROVIDED.

5,3 PRG2-RECEIVER SCOPE LOOP

THE PURPOSE OF PRG2 IS TO ALLOW SCOPING OF RECEIVER FUNCTIONS IN A RUN CONDITION USING USER SPECIFIED DL11-E PARAMETERS AND DATA. NO ERROR PRINTOUTS ARE PROVIDED.

5,4 PRG3-SINGLE CHARACTER MAINT MODE DATA TEST

PRG3 TRANSMITS, RECEIVES AND CHECKS RECEIVED DATA USING USER SPECIFIED DL11-E PARAMETERS, AND DATA.

5,4,1 ERROR PRINTOUTS

SELF EXPLANITORY ERROR PRINTOUTS ARE PROVIDED.

5,5 PRG4-SPECIAL BINARY COUNT MAINT MODE DATA TEST

PRG4 IS THE SAME AS PRG3 ROUTINE 54 EXCEPT THAT THE USER SPECIFIES DL11-E RUNNING PARAMETERS.

5,5,1 ERROR PRINTOUTS

SELF EXPLANATORY PRINTOUTS ARE PROVIDED.

6,0 POWER FAIL

A POWER FAIL ROUTINE IS INCLUDED IN THE PROGRAM. WHEN THE POWER FAILS THE PROGRAM WILL AUTOMATICALLY RESTART USING THE PRESENT SR OPTIONS AND THE LINE PREVIOUSLY SELECTED. NOTE: THE POWER MAY FAIL WHEN THE PROGRAM IS EXECUTING A 'WAIT' INSTRUCTION, IN THIS CASE OPERATOR INTERVENTION IS NEEDED TO PRESS CONTINUE, AN ERROR TYPEOUT RESULTS AND WILL TYPE THE PROGRAM #, THE ROUTINE THAT WAS RUNNING AT THE TIME THE POWER FAILED (PROGRAM 0 ONLY), AND THE PC OF THE POWER FAIL ERROR CALL.

RECOVERED FROM POWER FAILURE.
P(PRG#) T(ROUTINE #) PC = (ADDRESS OF ERROR CALL)

DLE1-E DATA

LINE # (CUST ID)	UNIT# (DEC ID)	RXCSR ADRS, RX VECTOR	FREQS, AVAILABLE	TELEPHONE NUMBER	REMARKS

```
                ;ENDR
                ;TITLE  DZDLAV12
                ;ABS
                ;PDP10
;
;DL11-E,C/D DIAGNOSTIC PROGRAM (OFF LINE TESTS)
;COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.
;
;PRG0= INPUT-OUTPUT LOGIC TESTS
;PRG1= TRANSMITTER SCOPE LOOP
;PRG2= RECEIVER SCOPE LOOP
;PRG3= SINGLE CHARACTER MAINTENANCE MODE DATA TEST
;PRG4= SPECIAL BINARY COUNT MAINTENANCE MODE DATA TEST
;
;STANDARD SR SWITCH OPTIONS (SWITCH SET TO A 1 )
;
;SR15= HALT ON ERROR
;SR14= SCOPE,
;SR13= INHIBIT PRINTOUT
;SR12= SELECT LINE NUMBER AND LOCK ON IT
;SR11= INHIBIT ITERATION,
;SR10= HALT AT END CURRENT TEST, TEST NO, IN DATA LIGHTS
;SR9= SELECT ROUTINE,
;SR7= DISABLE STALL MODE AND RUN FULL SPEED,
;SR6 THROUGH SR0 = NUMBER OF ROUTINE TO BE SELECTED,
;
;STANDARD CONFIGURATION
;CHARACTER LENGTH 8
;STOP CODE 2
```

```
                000000                  .=0
000000  005256                  ERTP                    ;UNASSIGNED TRAP
000002  000000                  0
000004  005256          MACHERR ERTP                    ;SP OVERFLOW, BUS ERROR TRAP
000006  000040                  40
000010  005256                  ERTP                    ;RESERVED INSTRUCTION TRAP
000012  000100                  100
000014  005256                  ERTP                    ;TRACE TRAP
000016  000140                  140
000020  005364                  MAPVEC                  ;TRAP TO MAP VECTOR
000022  000340                  PRTY7
000024  005604                  PFAIL                   ;POWER FAIL TRAP
000026  000340                  PRTY7
000030  002510                  EMTINT                  ;EMT TRAP
000032  000340                  PRTY7
000034  005256                  ERTP
000036  000340                  340
000040  000042                  .+2
000042  000000                  HALT
                                .REPT   119.
                                .+2                     ;TRAP TO TRAP REPORTER
                                4
                                .ENDR
000044  000046                  .+2                     ;TRAP TO TRAP REPORTER
000046  000004                  4
000050  000052                  .+2                     ;TRAP TO TRAP REPORTER
000052  000004                  4
000054  000056                  .+2                     ;TRAP TO TRAP REPORTER
000056  000004                  4
000060  000062                  .+2                     ;TRAP TO TRAP REPORTER
000062  000004                  4
000064  000066                  .+2                     ;TRAP TO TRAP REPORTER
000066  000004                  4
000070  000072                  .+2                     ;TRAP TO TRAP REPORTER
000072  000004                  4
000074  000076                  .+2                     ;TRAP TO TRAP REPORTER
000076  000004                  4
000100  000102                  .+2                     ;TRAP TO TRAP REPORTER
000102  000004                  4
000104  000106                  .+2                     ;TRAP TO TRAP REPORTER
000106  000004                  4
000110  000112                  .+2                     ;TRAP TO TRAP REPORTER
000112  000004                  4
000114  000116                  .+2                     ;TRAP TO TRAP REPORTER
000116  000004                  4
000120  000122                  .+2                     ;TRAP TO TRAP REPORTER
000122  000004                  4
000124  000126                  .+2                     ;TRAP TO TRAP REPORTER
000126  000004                  4
000130  000132                  .+2                     ;TRAP TO TRAP REPORTER
000132  000004                  4
000134  000136                  .+2                     ;TRAP TO TRAP REPORTER
000136  000004                  4
000140  000142                  .+2                     ;TRAP TO TRAP REPORTER
000142  000004                  4
```

```
000144  000146                  ;+2                 ;TRAP TO TRAP REPORTER
000146  000004                  4
000150  000152                  ;+2                 ;TRAP TO TRAP REPORTER
000152  000004                  4
000154  000156                  ;+2                 ;TRAP TO TRAP REPORTER
000156  000004                  4
000160  000162                  ;+2                 ;TRAP TO TRAP REPORTER
000162  000004                  4
000164  000166                  ;+2                 ;TRAP TO TRAP REPORTER
000166  000004                  4
000170  000172                  ;+2                 ;TRAP TO TRAP REPORTER
000172  000004                  4
000174  000176                  ;+2                 ;TRAP TO TRAP REPORTER
000176  000004                  4
000200  000202                  ;+2                 ;TRAP TO TRAP REPORTER
000202  000004                  4
000204  000206                  ;+2                 ;TRAP TO TRAP REPORTER
000206  000004                  4
000210  000212                  ;+2                 ;TRAP TO TRAP REPORTER
000212  000004                  4
000214  000216                  ;+2                 ;TRAP TO TRAP REPORTER
000216  000004                  4
000220  000222                  ;+2                 ;TRAP TO TRAP REPORTER
000222  000004                  4
000224  000226                  ;+2                 ;TRAP TO TRAP REPORTER
000226  000004                  4
000230  000232                  ;+2                 ;TRAP TO TRAP REPORTER
000232  000004                  4
000234  000236                  ;+2                 ;TRAP TO TRAP REPORTER
000236  000004                  4
000240  000242                  ;+2                 ;TRAP TO TRAP REPORTER
000242  000004                  4
000244  000246                  ;+2                 ;TRAP TO TRAP REPORTER
000246  000004                  4
000250  000252                  ;+2                 ;TRAP TO TRAP REPORTER
000252  000004                  4
000254  000256                  ;+2                 ;TRAP TO TRAP REPORTER
000256  000004                  4
000260  000262                  ;+2                 ;TRAP TO TRAP REPORTER
000262  000004                  4
000264  000266                  ;+2                 ;TRAP TO TRAP REPORTER
000266  000004                  4
000270  000272                  ;+2                 ;TRAP TO TRAP REPORTER
000272  000004                  4
000274  000276                  ;+2                 ;TRAP TO TRAP REPORTER
000276  000004                  4
000300  000302                  ;+2                 ;TRAP TO TRAP REPORTER
000302  000004                  4
000304  000306                  ;+2                 ;TRAP TO TRAP REPORTER
000306  000004                  4
000310  000312                  ;+2                 ;TRAP TO TRAP REPORTER
000312  000004                  4
000314  000316                  ;+2                 ;TRAP TO TRAP REPORTER
000316  000004                  4
000320  000322                  ;+2                 ;TRAP TO TRAP REPORTER
```

```
000322  000004                          4
000324  000326                          .+2             ;TRAP TO TRAP REPORTER
000326  000004                          4
000330  000332                          .+2             ;TRAP TO TRAP REPORTER
000332  000004                          4
000334  000336                          .+2             ;TRAP TO TRAP REPORTER
000336  000004                          4
000340  000342                          .+2             ;TRAP TO TRAP REPORTER
000342  000004                          4
000344  000346                          .+2             ;TRAP TO TRAP REPORTER
000346  000004                          4
000350  000352                          .+2             ;TRAP TO TRAP REPORTER
000352  000004                          4
000354  000356                          .+2             ;TRAP TO TRAP REPORTER
000356  000004                          4
000360  000362                          .+2             ;TRAP TO TRAP REPORTER
000362  000004                          4
000364  000366                          .+2             ;TRAP TO TRAP REPORTER
000366  000004                          4
000370  000372                          .+2             ;TRAP TO TRAP REPORTER
000372  000004                          4
000374  000376                          .+2             ;TRAP TO TRAP REPORTER
000376  000004                          4
000400  000402                          .+2             ;TRAP TO TRAP REPORTER
000402  000004                          4
000404  000406                          .+2             ;TRAP TO TRAP REPORTER
000406  000004                          4
000410  000412                          .+2             ;TRAP TO TRAP REPORTER
000412  000004                          4
000414  000416                          .+2             ;TRAP TO TRAP REPORTER
000416  000004                          4
000420  000422                          .+2             ;TRAP TO TRAP REPORTER
000422  000004                          4
000424  000426                          .+2             ;TRAP TO TRAP REPORTER
000426  000004                          4
000430  000432                          .+2             ;TRAP TO TRAP REPORTER
000432  000004                          4
000434  000436                          .+2             ;TRAP TO TRAP REPORTER
000436  000004                          4
000440  000442                          .+2             ;TRAP TO TRAP REPORTER
000442  000004                          4
000444  000446                          .+2             ;TRAP TO TRAP REPORTER
000446  000004                          4
000450  000452                          .+2             ;TRAP TO TRAP REPORTER
000452  000004                          4
000454  000456                          .+2             ;TRAP TO TRAP REPORTER
000456  000004                          4
000460  000462                          .+2             ;TRAP TO TRAP REPORTER
000462  000004                          4
000464  000466                          .+2             ;TRAP TO TRAP REPORTER
000466  000004                          4
000470  000472                          .+2             ;TRAP TO TRAP REPORTER
000472  000004                          4
000474  000476                          .+2             ;TRAP TO TRAP REPORTER
000476  000004                          4
```

```
000500  000502          ;+2         ;TRAP TO TRAP REPORTER
000502  000004          4
000504  000506          ;+2         ;TRAP TO TRAP REPORTER
000506  000004          4
000510  000512          ;+2         ;TRAP TO TRAP REPORTER
000512  000004          4
000514  000516          ;+2         ;TRAP TO TRAP REPORTER
000516  000004          4
000520  000522          ;+2         ;TRAP TO TRAP REPORTER
000522  000004          4
000524  000526          ;+2         ;TRAP TO TRAP REPORTER
000526  000004          4
000530  000532          ;+2         ;TRAP TO TRAP REPORTER
000532  000004          4
000534  000536          ;+2         ;TRAP TO TRAP REPORTER
000536  000004          4
000540  000542          ;+2         ;TRAP TO TRAP REPORTER
000542  000004          4
000544  000546          ;+2         ;TRAP TO TRAP REPORTER
000546  000004          4
000550  000552          ;+2         ;TRAP TO TRAP REPORTER
000552  000004          4
000554  000556          ;+2         ;TRAP TO TRAP REPORTER
000556  000004          4
000560  000562          ;+2         ;TRAP TO TRAP REPORTER
000562  000004          4
000564  000566          ;+2         ;TRAP TO TRAP REPORTER
000566  000004          4
000570  000572          ;+2         ;TRAP TO TRAP REPORTER
000572  000004          4
000574  000576          ;+2         ;TRAP TO TRAP REPORTER
000576  000004          4
000600  000602          ;+2         ;TRAP TO TRAP REPORTER
000602  000004          4
000604  000606          ;+2         ;TRAP TO TRAP REPORTER
000606  000004          4
000610  000612          ;+2         ;TRAP TO TRAP REPORTER
000612  000004          4
000614  000616          ;+2         ;TRAP TO TRAP REPORTER
000616  000004          4
000620  000622          ;+2         ;TRAP TO TRAP REPORTER
000622  000004          4
000624  000626          ;+2         ;TRAP TO TRAP REPORTER
000626  000004          4
000630  000632          ;+2         ;TRAP TO TRAP REPORTER
000632  000004          4
000634  000636          ;+2         ;TRAP TO TRAP REPORTER
000636  000004          4
000640  000642          ;+2         ;TRAP TO TRAP REPORTER
000642  000004          4
000644  000646          ;+2         ;TRAP TO TRAP REPORTER
000646  000004          4
000650  000652          ;+2         ;TRAP TO TRAP REPORTER
000652  000004          4
000654  000656          ;+2         ;TRAP TO TRAP REPORTER
```

```
000656  000004                          4
000660  000662                          .+2     ;TRAP TO TRAP REPORTER
000662  000004                          4
000664  000666                          .+2     ;TRAP TO TRAP REPORTER
000666  000004                          4
000670  000672                          .+2     ;TRAP TO TRAP REPORTER
000672  000004                          4
000674  000676                          .+2     ;TRAP TO TRAP REPORTER
000676  000004                          4
000700  000702                          .+2     ;TRAP TO TRAP REPORTER
000702  000004                          4
000704  000706                          .+2     ;TRAP TO TRAP REPORTER
000706  000004                          4
000710  000712                          .+2     ;TRAP TO TRAP REPORTER
000712  000004                          4
000714  000716                          .+2     ;TRAP TO TRAP REPORTER
000716  000004                          4
000720  000722                          .+2     ;TRAP TO TRAP REPORTER
000722  000004                          4
000724  000726                          .+2     ;TRAP TO TRAP REPORTER
000726  000004                          4
000730  000732                          .+2     ;TRAP TO TRAP REPORTER
000732  000004                          4
000734  000736                          .+2     ;TRAP TO TRAP REPORTER
000736  000004                          4
000740  000742                          .+2     ;TRAP TO TRAP REPORTER
000742  000004                          4
000744  000746                          .+2     ;TRAP TO TRAP REPORTER
000746  000004                          4
000750  000752                          .+2     ;TRAP TO TRAP REPORTER
000752  000004                          4
000754  000756                          .+2     ;TRAP TO TRAP REPORTER
000756  000004                          4
000760  000762                          .+2     ;TRAP TO TRAP REPORTER
000762  000004                          4
000764  000766                          .+2     ;TRAP TO TRAP REPORTER
000766  000004                          4
000770  000772                          .+2     ;TRAP TO TRAP REPORTER
000772  000004                          4
000774  000776                          .+2     ;TRAP TO TRAP REPORTER
000776  000004                          4
```

```
                        ;EQUATE STATEMENTS
177570                  SR=177570
177776                  PSW=177776
001176                  SPBOT=1176
000240                  NOP=240
000000                  OPEN=0
100000                  MANUAL=BIT15
100000                  BIT15=100000
040000                  BIT14=40000
020000                  BIT13=20000
010000                  BIT12=10000
004000                  BIT11=4000
002000                  BIT10=2000
001000                  BIT9=1000
000400                  BIT8=400
000200                  BIT7=200
000100                  BIT6=100
000040                  BIT5=40
000020                  BIT4=20
000010                  BIT3=10
000004                  BIT2=4
000002                  BIT1=2
000001                  BIT0=1
005726                  POPSP=5726              ;POP THE STACK, SAME AS TST (6)+
022626                  POPSP2=022626           ;POP STACK TWICE, SAME AS CMP (6)+,(6)+
000340                  PRTY7=340               ;PRIORITY LEVEL DEFINITIONS
000300                  PRTY6=300
000240                  PRTY5=240
000200                  PRTY4=200
000140                  PRTY3=140
000100                  PRTY2=100
000040                  PRTY1=40
000000                  PRTY0=0
104000                  TYPE=EMT+0
104001                  TYPES=EMT+1
104002                  STALL=EMT+2
104003                  ERROR=EMT+3
104004                  DATCH=EMT+4
104005                  CMALT=EMT+5
104006                  STPXV=EMT+6
104007                  SYSXV=EMT+7
104010                  ERRAL=EMT+10
104011                  CRESET=EMT+11
104012                  SCOPE=EMT+12
104013                  SAVREG=EMT+13
104014                  HSTREG=EMT+14
104015                  ERRCRL=EMT+15
104016                  DELAY=EMT+16
104017                  TIMERX=EMT+17
104020                  TIMETX=EMT+20
177777                  ATLAST=-1
100000                  CD=100000               ;FLAG FOR C/D TESTS
```

```
                000200
000200  000200  001430                  .=200
                000204                  JMP     START           ;GO TO START OF PROGRAM.
000204  000167  005542                  .=204
                000210                  JMP     @RESTART
000210  000167  001414                  .=210
                001200                  JMP     REMAP           ;REMAP AND START
                                        .=1200
                                ;
                                ;DEVICE ADDRESS LIST
                                ;LSB BIT@ IS SET TO A 1 BY MAPPER IF DEVICE NOT FOUND
                                ;TO TEST THAT LINE NOT FOUND CLEAR BIT@ IN THAT DEVICE ADDRESS
                                ;IN THIS TABLE AFTER MAPPING DONE
                                ;*********************
001200  175610                  RXCR0:  175610          ;LINE 0 DEVICE ADDRESS (RXCSR)
001202  175620                  RXCR1:  175620          ;LINE 1 DEVICE ADDRESS (RXCSR)
001204  175630                  RXCR2:  175630          ;LINE 2 DEVICE ADDRESS (RXCSR)
001206  175640                  RXCR3:  175640          ;LINE 3 DEVICE ADDRESS (RXCSR)
001210  175650                  RXCR4:  175650          ;LINE 4 DEVICE ADDRESS (RXCSR)
001212  175660                  RXCR5:  175660          ;LINE 5 DEVICE ADDRESS (RXCSR)
001214  175670                  RXCR6:  175670          ;LINE 6 DEVICE ADDRESS (RXCSR)
001216  175700                  RXCR7:  175700          ;LINE 7 DEVICE ADDRESS (RXCSR)
001220  175710                  RXCR10: 175710          ;LINE 10 DEVICE ADDRESS (RXCSR)
001222  175720                  RXCR11: 175720          ;LINE 11 DEVICE ADDRESS (RXCSR)
001224  175730                  RXCR12: 175730          ;LINE 12 DEVICE ADDRESS (RXCSR)
001226  175740                  RXCR13: 175740          ;LINE 13 DEVICE ADDRESS (RXCSR)
001230  175750                  RXCR14: 175750          ;LINE 14 DEVICE ADDRESS (RXCSR)
001232  175760                  RXCR15: 175760          ;LINE 15 DEVICE ADDRESS (RXCSR)
001234  175770                  RXCR16: 175770          ;LINE 16 DEVICE ADDRESS (RXCSR)
001236  176000                  RXCR17: 176000          ;LINE 17 DEVICE ADDRESS (RXCSR)
001240  176010                  RXCR20: 176010          ;LINE 20 DEVICE ADDRESS (RXCSR)
001242  176020                  RXCR21: 176020          ;LINE 21 DEVICE ADDRESS (RXCSR)
001244  176030                  RXCR22: 176030          ;LINE 22 DEVICE ADDRESS (RXCSR)
001246  176040                  RXCR23: 176040          ;LINE 23 DEVICE ADDRESS (RXCSR)
001250  176050                  RXCR24: 176050          ;LINE 24 DEVICE ADDRESS (RXCSR)
001252  176060                  RXCR25: 176060          ;LINE 25 DEVICE ADDRESS (RXCSR)
001254  176070                  RXCR26: 176070          ;LINE 26 DEVICE ADDRESS (RXCSR)
001256  176100                  RXCR27: 176100          ;LINE 27 DEVICE ADDRESS (RXCSR)
001260  176110                  RXCR30: 176110          ;LINE 30 DEVICE ADDRESS (RXCSR)
001262  176120                  RXCR31: 176120          ;LINE 31 DEVICE ADDRESS (RXCSR)
001264  176130                  RXCR32: 176130          ;LINE 32 DEVICE ADDRESS (RXCSR)
001266  176140                  RXCR33: 176140          ;LINE 33 DEVICE ADDRESS (RXCSR)
001270  176150                  RXCR34: 176150          ;LINE 34 DEVICE ADDRESS (RXCSR)
001272  176160                  RXCR35: 176160          ;LINE 35 DEVICE ADDRESS (RXCSR)
001274  176170                  RXCR36: 176170          ;LINE 36 DEVICE ADDRESS (RXCSR)
001276  177777                  RXEND:  177777          ;LINE XX DEVICE ADDRESS (RXCSR)
```

```
;
;CHARACTER LENGTH, PRIORITY, C/D MASK
;INITIALLY SET FOR DL11-E, PRIORITY=4,CHARACTER LENGTH=8
;BIT 15 SET TO A 1 = THAT LINE HAS DL11-D OR DL11-C
;EX: 140377 = DL11C OR DL11D, PRIORITY = 5, CHARACTER LENGTH = 8
;BITS 12-14 = PRIORITY LEVEL THAT LINE
;BITS 0-7 = CHARACTER MASK (I.E. 377=8, 177,7, 77=6, 37=5
;*********************
CMAS0:  140377          ;LINE 0 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS1:  140377          ;LINE 1 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS2:  140377          ;LINE 2 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS3:  140377          ;LINE 3 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS4:  140377          ;LINE 4 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS5:  140377          ;LINE 5 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS6:  140377          ;LINE 6 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS7:  140377          ;LINE 7 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS10: 140377          ;LINE 10 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS11: 140377          ;LINE 11 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS12: 140377          ;LINE 12 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS13: 140377          ;LINE 13 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS14: 140377          ;LINE 14 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS15: 140377          ;LINE 15 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS16: 140377          ;LINE 16 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS17: 140377          ;LINE 17 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS20: 140377          ;LINE 20 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS21: 140377          ;LINE 21 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS22: 140377          ;LINE 22 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS23: 140377          ;LINE 23 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS24: 140377          ;LINE 24 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS25: 140377          ;LINE 25 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS26: 140377          ;LINE 26 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS27: 140377          ;LINE 27 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS30: 140377          ;LINE 30 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS31: 140377          ;LINE 31 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS32: 140377          ;LINE 32 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS33: 140377          ;LINE 33 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS34: 140377          ;LINE 34 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS35: 140377          ;LINE 35 CHARACTER MASK, PRIORITY, C/D FLAG
CMAS36: 140377          ;LINE 36 CHARACTER MASK, PRIORITY, C/D FLAG

001300  140377
001302  140377
001304  140377
001306  140377
001310  140377
001312  140377
001314  140377
001316  140377
001320  140377
001322  140377
001324  140377
001326  140377
001330  140377
001332  140377
001334  140377
001336  140377
001340  140377
001342  140377
001344  140377
001346  140377
001350  140377
001352  140377
001354  140377
001356  140377
001360  140377
001362  140377
001364  140377
001366  140377
001370  140377
001372  140377
001374  140377
```

```
                        ;
001376  000000         UMASK:  0                       ;MASK FOR DEVICE UT
001400  000000         RMASK:  0                       ;MASK FOR CHAR LENGTH FOR D[illegible]E UT
001402  177740         STLMSK: 177740                  ;MASK FOR MAX RANDOM STALL
                        ;
001404  000000         RXCSR:  0                       ;RECEIVER UNDER TEST
001406  000000         RXBUF:  0                       ;RECEIVER BUFFER UNDER TEST
001410  000000         TXCSR:  0                       ;TRANSMITTER CSR UNDER TEST
001412  000000         TXBUF:  0                       ;TRANSMITTER BUFFER UNDER TEST
001414  000000         RXVTR:  0                       ;RECEIVER VECTOR UNDER TEST
001416  000000         RXLVL:  0                       ;RECEIVER PRIORITY LE[illegible]L UT
001420  000000         TXVTR:  0                       ;TRANSMITTER VECTOR U[illegible]ER TEST
001422  000000         TXLVL:  0                       ;TRANSMITTER PRIORITY LEVEL UT
                        ;**********************
                        ;
001424  177560         TKS:    177560                  ;LSR CSR
001426  177562         TKB:    177562                  ;LSR BUFFER
001430  177564         TPS:    177564                  ;LSP CSR
001432  177566         TPB:    177566                  ;LSP BUFFER
001434  000060         TKVTR:  60                      ;LSR INTERRUPT VECTOR
001436  000200         TKLVL:  PRTY4                   ;LSR PRIORITY LEVEL
001440  000064         TPVTR:  64                      ;LSP INTERRUPT VECTOR
001442  000200         TPLVL:  PRTY4                   ;LSP PRIORITY LEVEL
001444  000000         PRGNUM: OPEN                    ;CONTAINS CURRENT PROGRAM#
001446  000000         KSTART: OPEN                    ;CURRENT PROGRAM START ADDRESS,
001450  000000         CURTST: OPEN                    ;CONTAINS ADDR OF CURRENT TEST,
001452  000000         RTNNO:  OPEN                    ;CONTAINS CURRENT TEST #,
001454  000000         TNNO:   0                       ;CONTAINES EDITED TNUM
001456  000000         NXTST:  OPEN                    ;CONTAINS ADDR OF NEXT TEST,
001460  000000         ICTR:   OPEN                    ;CONTAINS CURRENT ITERATION COUNT
001462  000000         SCOPTR: OPEN                    ;CONTAINS CURRENT SCOPE POINTER,
001464  000000         OLDPS:  0                       ;PS SAVED FROM TRAP TO ENT ROUTINE
001466  000000         FMAP:   0                       ;MAPPING FLAG, 1= MAPPING IN PROGRESS
001470  006004         PRGTAB: PRG0                    ;PRG0 START ADDRESS
001472  013504                 PRG1                    ;PRG1 START ADDRESS
001474  013544                 PRG2                    ;PRG2 START ADDRESS
001476  013630                 PRG3                    ;PRG3 START ADDRESS
001500  013662                 PRG4                    ;PRG4 START ADDRESS
001502  004460                 INCRPG                  ;INCORRECT PROGRAM SELECTED
001504  004460                 INCRPG
001506  004460                 INCRPG
```

```
001510  003064          EMTTAB: TYP             ;POINTER TO TYPEOUT ROUTINE
001512  003206                  SYS             ;POINTER TO CHAINED MESSAGES ROUTINE
001514  003344                  STAL            ;POINTER TO RANDOM STALL ROUTINE
001516  005066                  ERR             ;POINTER TO ERROR ROUTINE
001520  004774                  DTCHK
001522  000000                  OPEN
001524  002642                  STLSRV
001526  002676                  STLSRV
001530  004732                  EHLT            ;POINTER TO ERROR HALT ROUTINE;
001532  002732                  SRSETT
001534  002322                  CHAINN
001536  002542                  SAVRG
001540  002602                  RSTRG
001542  005110                  ERR1
001544  003276                  DLY
001546  003414                  TMRX
001550  003424                  TMTX
                        ;
001552  000000          CRBUF:  OPEN
001554  000000          CRBUFA: OPEN
001556  000000          CRBUFB: OPEN
001560  000000          CTR0:   OPEN
001562  000000          CTR1:   OPEN
001564  000000          CTR2:   OPEN
001566  000000          CTR3:   OPEN
001570  000000          CTR4:   OPEN
001572  000000          CTR5:   OPEN
001574  000000          CTR6:   OPEN
001576  000000          CTR7:   OPEN
001600  000000          TXCSRT: OPEN
001602  000000          RXCSRT: OPEN
001604  000000          RXBUFT: OPEN
001606  000000          FCUNDV: 0
001610  000000          LINENO: 0
001612  000000          TEMP:   OPEN
001614  000000          TEMP1:  0
001616  000000          FTITLE: 0
001620  000000          FNONE:  0
001622  000000          TOPC:   0
001624  000000          FROMPC: 0
001626  000000          PASCNT: 0
```

```
                ;MACR   TSTAA   AX,B,C,D,E
;*******************************************************************************
AT'E'!  C                       ;TEST NUMBER                                    *
        AT'D'                   ;ADDRESS OF NEXT TEST                           *
        B                       ;ITERATION COUNT                                *
        'AX'A                   ;SCOPE ENTRY POINT                              *
        X=X+1                   ;                                               *
;*******************************************************************************
                ;ENDM
                ;MACR   TSTA    B,AX,Z
                TSTAA   AX,B,\X+1+2,\X+2,\X+1
                ;ENDM
```

```
001630  005067  177762          REMAP:  CLR     FTITLE          ;CLEAR MAP FLAG
001634  012706  001170          START:  MOV     #SPBOT,SP       ;SET BOTTOM OF SP STACK.
001640  012767  005004  176156          MOV     #PFAIL,24
001646  005067  177734                  CLR     FOUNDV
001652  005067  177610                  CLR     FMAP
001656  004767  001140                  JSR     R7,CLRCD        ;CLEAR DEVICE UT PARAMETERS
001662  004767  001352                  JSR     R7,OVRLAY       ;OVERLAY MAP AREA
001666  005767  177724                  TST     FTITLE          ;TITLE PRINTED AND MAP MADE
001672  001054                          BNE     START1          ;YES, SKIP OVER THIS
001674  104000                          TYPE
001676  013770                          MTIT
001700  005267  177712                  INC     FTITLE
001704  005067  177710                  CLR     FNONE           ;CLEAR DEVICE PRESENT FLAG
001710  012767  001750  176066          MOV     #MAPNE,MACRER   ;SET UP NO DEVICE PRESENT RETURN
001716  012704  001200                  MOV     #RMXCSR,R4      ;SET UP DEVICE POINTER
001722  020427  001276          MAPA:   CMP     R4,#RMXEND      ;LAST DEVICE
001726  001430                          BEQ     MAPEND          ;YES, EXIT
001730  042714  000001                  BIC     #BIT0,(4)       ;CLEAR ODD ADDRESS
001734  005067  176036                  CLR     PSW
001740  005774  000000                  TST     @(4)            ;TEST DEVICE
001744  000240                          NOP
001746  000404                          BR      MAPOK
001750  052724  000001          MAPNE:  BIS     #BIT0,(4)+      ;NOT LIVING
001754  022626                          POPSP2
001756  000761                          BR      MAPA
001760  012467  177630          MAPOK:  MOV     (4)+,TEMP1      ;SAVE DEVICE ADDRESS FOR TYPING
001764  004567  002054                  JSR     R5,OCONV
001770  001614                          TEMP1
001772  014100                          MDEVAD
001774  000006                          6
001776  104000                          TYPE
002000  014100                          MDEVAD
002002  005267  177612                  INC     FNONE           ;SET HAVE DEVICE
002006  000745                          BR      MAPA
002010  012767  005256  175766  MAPEND: MOV     #ERTP,MACRER    ;RESET TRAPS
002016  005767  177576                  TST     FNONE           ;ANY DEVICES PRESENT
002022  001424                          BEQ     NOPERR          ;NO, ERROR
002024  012701  001200          START1: MOV     #RMXCSR,R1
002030  032711  000001          START2: BIT     #BIT0,(1)       ;IS DEVICE LIVING
002034  001013                          BNE     START3          ;NO, CHECK FOR END
002036  010167  177546                  MOV     R1,LINENO       ;CALCULATE LINE NUMBER UNDER TEST
002042  162767  001200  177540          SUB     #RMXCSR,LINENO
002050  006267  177534                  ASR     LINENO
002054  011101                          MOV     (1),R1          ;YES, LOAD AND EXIT
002056  004767  003360                  JSR     R7,FORMAT
002062  000420                          BR      START4
```

```
002064  005721                          START3: TST     (1)+
002066  020127  001276                          CMP     R1,#RXEND       ;END OF TABLE
002072  001356                              BNE     START2          ;NO, LOOP
002074  104000                          MAPERR: TYPE
002076  014120                              MNONE
002100  005737  000042                      TST     @#42            ;MONITOR LOAD
002104  001402                              BEQ     .+6             ;NO, CONTINUE
002106  000167  002540                      JMP     PRGXTL          ;YES, EXIT
002112  005067  177500                      CLR     PTITLE
002116  000000                              HALT
002120  000167  177510                      JMP     START
002124  012767  000001  177474    START4: MOV     #1,PASCNT
002132  005067  175640                      CLR     PSW
002136  005067  177310                      CLR     RTNNO
002142  016700  175422                      MOV     SR,R0           ;(SR) TO R0
002146  042700  177770                      BIC     #177770,R0      ;LIMIT (SR) TO BITS 3-0
002152  010067  177266                      MOV     R0,PRGNUM       ;SAVE PROGRAM #
002156  006300                              ASL     R0
002160  000170  001470                      JMP     @PRGTAB(R0)     ;GO TO SELECTED PROGRAM
                                        ;
002164  016767  177256  177264  GETRDY: MOV     KSTART,NXTST    ;ADDR OF 1ST ROUTINE TO NXTST
002172  012767  005256  175604  GTRDYX: MOV     #ERTP,MACHER    ;RESET MACHER TRAP
002200  012767  000040  175600          MOV     #40,MACHER+2
002206  005067  177254                      CLR     PMAP
002212  012706  001176                      MOV     #SPBOT,R6       ;SET BOTTOM OF STACK
002216  104011                              RESET                   ;ISSUE RESET
002220  005067  175552                      CLR     PSW
002224  004767  000226          GTRDYA: JSR     R7,FORWD        ;ROLL FORWARD TO "NEXT" ROUTINE
002230  032767  001000  175332              BIT     #BIT9,SR        ;CHECK SELECT ROUTINE SWITCH
002236  001011                              BNE     GTRDYC          ;BRANCH IF SELECT ROUTINE SWITCH IS SET
002240  005767  177132                      TST     UMASK           ;C/D DEVICE
002244  100003                              BPL     GTRDA1          ;NO, CONTINUE
002246  005767  177200                      TST     RTNNO           ;THIS A C/D TEST
002252  100364                              BPL     GTRDYA          ;NO, DO NEXT TEST
002254  000177  177170          GTRDA1: JMP     @CURTST         ;GO RUN CURRENT ROUTINE
002260  000450                              BR      CHNG            ;NO GO, MANUAL RTN BYPASSED
002262  016700  175302          GTRDYC: MOV     SR,R0           ;(SR) TO R0
002266  042780  177000                      BIC     #177000,R0      ;MASK UNDESIRED BITS
002272  126700  177154                      CMPB    RTNNO,R0        ;COMPARE RTNNO TO (R0)
002276  001002                              BNE     GTRDYD          ;BRANCH IF ROUTINE NOT FOUND YET
002300  000177  177144                      JMP     @CURTST         ;GO RUN ROUTINE
002304  022767  177777  177144  GTRDYD: CMP     #-1,NXTST       ;NO, CHECK FOR LAST ROUTINE
002312  001344                              BNE     GTRDYA          ;BRANCH IF NOT LAST ROUTINE
002314  004767  002130                      JSR     R7,INCRTN       ;YES, INCORRECT ROUTINE SELECTED
002320  000721                              BR      GETRDY          ;START OVER
```

```
                                        ;
002322  032767  040000  175240  CHAINN: BIT     #BIT14,SR       ;CHECK FOR SCOPE OPTION,
002330  001403                          BEQ     CHNA            ;BRANCH IF SCOPE SW NOT SET,
002332  016716  177124          CHNAB:  MOV     SCOPTR,@X6      ;SET UP TO RETURN TO ROUTINE,
002336  000002                          RTI                     ;RETURN TO ROUTINE,
002340  032767  004000  175222  CHNA:   BIT     #BIT11,SR       ;TEST INHIBIT ITERATION SWITCH
002346  001003                          BNE     CHNAA           ;BRANCH IF INHIBIT ITERATION SW SET,
002350  005367  177104                  DEC     ICTR            ;DECREMENT ITERATION COUNT,
002354  001366                          BNE     CHNAB           ;BRANCH IF COUNT NOT 0,
002356  022626                  CHNAA:  POPSP2                  ;POP STACK TWICE
002360  032767  002000  175202          BIT     #BIT10,SR
002366  001405                          BEQ     CHNB
002370  016700  177056                  MOV     RTNNO,X0
002374  042700  100000                  BIC     #BIT15,X0
002400  000000                          HALT
002402  032767  001000  175160  CHNB:   BIT     #BIT9,SR        ;CHECK SELECT ROUTINE SWITCH
002410  001265                          BNE     GETRDY          ;BRANCH IF SELECT RTN SW SET
002412  022767  177777  177036          CMP     #-1,NXTGT       ;LAST TEST?
002420  001264                          BNE     GTRDYX          ;BRANCH IF NOT LAST TEST,
002422  004767  002044                  JSR     X7,PRGEND       ;PROGRAM END,
002426  000696                          BR      GETRDY
                                        ;
                                        ;INIT FOR C/D = WITHOUT JUMPER RESET STARTS ASSEMBLING CHARACTER SETTING DONE
                                        ;SET MAINT, DELAY, CLEAR RX DONE
002430  005767  176742          CDINIT: TST     UMASK           ;C&D DEVICE
002434  100007                          BPL     CDINX           ;NO, EXIT
002436  052777  000004  176744          BIS     #BIT2,@TXCSR    ;SET MAINT BIT
002444  104016                          DELAY                   ;WAIT 1,5 SEC
002446  002734                          1500.
002450  005777  176732                  TST     @RXBUF          ;CLEAR RX DONE
002454  000207                  CDINX:  RTS     X7
```

```
                                        ;
002456  016705  176774        FORWD:  MOV     NXTST,R5        ;ADDR OF NEXT ROUTINE TO R5,
002462  012567  176764                MOV     (R5)+,RTNNO     ;GET NEXT ROUTINE NUMBER,
002466  012567  176764                MOV     (R5)+,NXTST     ;GET ADDR OF NEXT "NEXT" ROUTINE,
002472  012567  176762                MOV     (R5)+,ICTR      ;GET ITERATION COUNT,
002476  012567  176760                MOV     (R5)+,SCOPTR    ;GET SCOPE LOOP ENTRY POINTER,
002502  010567  176742                MOV     R5,CURTST       ;ADDR OF NOW CURRENT TEST TO CURTST,
002506  000207                        RTS     R7              ;EXIT FORWD SUBROUTINE;
                                        ;
002510  011646        EMTINT: MOV     @R6,-(SP)       ;GET SAVED PC,
002512  162716  000002                SUB     #2,@R6          ;DECREMENT PC BY 2;
002516  017616  000000                MOV     @(SP),@R6
002522  006316        EMTA:   ASL     @R6             ;EMT ARG X 2,
002524  042716  177001                BIC     #177001,@R6     ;REMOVE 7 MSB,
002530  062716  001510                ADD     #EMTTAB,@R6     ;FORM EMT RTN ADDR;
002534  017616  000000                MOV     @(SP),@R6
002540  000136                        JMP     @(SP)+          ;GO TO EMT ROUTINE;
                                        ;
                                        ;SAVE REGS 0 TO 4 SUBROUTINE;
                                        ;
002542  012667  000030        SAVRG:  MOV     (SP)+,SVRPC     ;SAVE PC AND PSW,
002546  012667  000026                MOV     (SP)+,SVRPSW
002552  010446                        MOV     R4,-(SP)        ;SAVE REGS 0 - 4
002554  010346                        MOV     R3,-(SP)        ;IN STACK;
002556  010246                        MOV     R2,-(SP)
002560  010146                        MOV     R1,-(SP)
002562  010046                        MOV     R0,-(SP)
002564  016746  000010                MOV     SVRPSW,-(SP)    ;RESTORE PC AND PSW,
002570  016746  000002                MOV     SVRPC,-(SP)
002574  000002                        RTI                     ;EXIT;
002576  000000        SVRPC:  OPEN
002600  000000        SVRPSW: OPEN
                                        ;
                                        ;RESTORE REGS 0 TO 4 SUBROUTINE,
                                        ;
002602  012667  000030        RSTRG:  MOV     (SP)+,RSTPC     ;SAVE PC AND PSW,
002606  012667  000026                MOV     (SP)+,RSTPSW
002612  012600                        MOV     (SP)+,R0        ;RESTORE REGS 0 - 4
002614  012601                        MOV     (SP)+,R1        ;FROM STACK,
002616  012602                        MOV     (SP)+,R2
002620  012603                        MOV     (SP)+,R3
002622  012604                        MOV     (SP)+,R4
002624  016746  000010                MOV     RSTPSW,-(SP)    ;RESTORE PC AND PSW,
002630  016746  000002                MOV     RSTPC,-(SP)
002634  000002                        RTI                     ;EXIT
002636  000000        RSTPC:  OPEN
002640  000000        RSTPSW: OPEN
```

```
                                        ;ROUTINE TO SET RECEIVER INTERRUPT VECTOR AND PRIORITY
002642  004767  002772          STLSPV: JSR     %7,TSTVEC
002646  017667  000000  000012          MOV     @(6),STPPA+2    ;MOVE VECTOR ADDR TO STPPA+2
002654  062716  000002                  ADD     #2,@%6          ;SET UP EXIT
002660  016701  176530                  MOV     RXVTR,%1
002664  012721  000000          STPPA:  MOV     #OPEN,(1)+      ;SET VECTOR ADDRESS
002670  016721  176522                  MOV     RXLVL,(1)+      ;SET PRIORITY
002674  000002                          RTI                     ;EXIT
                                        ;
                                        ;ROUTINE TO SET TRANSMITTER INTERRUPT VECTOR AND PRIORITY;
002676  004767  002736          STLSPV: JSR     %7,TSTVEC
002702  017667  000000  000012          MOV     @(6),STPPA+2    ;MOVE VECTOR ADDR TO STPPA+2
002710  062716  000002                  ADD     #2,@%6          ;SET UP EXIT
002714  016701  176500                  MOV     TXVTR,%1
002720  012721  000000          STPPA:  MOV     #OPEN,(1)+      ;SET VECTOR ADDRESS;
002724  016721  176472                  MOV     TXLVL,(1)+      ;SET PRIORITY
002730  000002                          RTI                     ;EXIT;
                                        ;
                                        ;ROUTINE TO ISSUE RESET;
002732  012700  052525          SRSETT: MOV     #52525,%0       ;DATA TO R0;
002736  005100                          COM     %0              ;COMPLEMENT (R0);
002740  010067  177770                  MOV     %0,SRSETT+2     ;(R0) TO SRSETT+2;
002744  000005                          RESET                   ;ISSUE RESET, (R0) IS
002746  000002                          RTI                     ;DISPLAYED; EXIT;
                                        ;
                                        ;RANDOM NUMBER GENERATOR, ROUTINE EXITS WITH NUMBER IN REGISTER 0;
002750  016700  000042          RNGEN:  MOV     RP1,%0
002754  006100                          ROL     %0
002756  006100                          ROL     %0
002760  066700  000034                  ADD     RP2,%0
002764  010067  000026                  MOV     %0,RP1
002770  006100                          ROL     %0
002772  006100                          ROL     %0
002774  066700  000020                  ADD     RP2,%0
003000  006100                          ROL     %0
003002  006100                          ROL     %0
003004  010067  000010                  MOV     %0,RP2
003010  016700  000002                  MOV     RP1,%0
003014  000207                          RTS     %7              ;EXIT; NUMBER IN R0
003016  001233          RP1:    1233
003020  007622          RP2:    7622
                                        ;
                                        ;CLRCD - CLEAR CURRENT DEVICE PARAMETERS
003022  005067  176364          CLRCD:  CLR     TXBUF
003026  005067  176356                  CLR     TXCSR
003032  005067  176346                  CLR     RXCSR
003036  005067  176344                  CLR     RXBUF
003042  005067  176346                  CLR     RXVTR
003046  005067  176346                  CLR     TXVTR
003052  005067  176340                  CLR     RXLVL
003056  005067  176340                  CLR     TXLVL
003062  000207                          RTS     %7
```

```
                                        ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
003064  011620                  TYP:    MOV     @%0,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS,
003066  062716  000002                  ADD     #2,@%6          ;SET UP EXIT,
003072  011000                          MOV     @%0,%0          ;ADDRESS OF MESSAGE TO R0,
003074  112067  000104          TYPA:   MOVB    (%)+,TYPDAT     ;GET CHARACTER
003100  122767  000100  000076          CMPB    #100,TYPDAT     ;CHECK FOR"@"CHARACTER
003106  001001                          BNE     TYPC            ;BRANCH IF NOT"@",
003110  000002                          RTI                     ;TERMINATOR CHAR, DONE, EXIT,
003112  122767  000045  000064  TYPC:   CMPB    #45,TYPDAT      ;CHECK FOR"%",
003120  001416                          BEQ     TYPF            ;BRANCH IF"%"
003122  122767  000043  000054          CMPB    #43,TYPDAT      ;NOT"%",CHECK FOR"#",
003130  001417                          BEQ     TYPG            ;BRANCH IF "#"
003132  004767  000002                  JSR     %7,TYPD         ;TYPE CHAR IN TYPDAT
003136  000756                          BR      TYPA
003140  116777  000040  176264  TYPD:   MOVB    TYPDAT,@TPB     ;OUTPUT CHARACTER TO PRINTER
003146  105777  176256                  TSTB    @TPS            ;WAIT FOR DONE FLAG,
003152  100375                          BPL     .-4
003154  000207                          RTS     %7              ;EXIT
003156  112767  000015  000020  TYPF:   MOVB    #15,TYPDAT      ;MOVE CARRIAGE RETURN CODE TO TYPDAT
003164  004767  177750                  JSR     %7,TYPD         ;GO TYPE CHAR,
003170  112767  000012  000006  TYPG:   MOVB    #12,TYPDAT      ;MOVE LF CODE TO TYPDAT,
003176  004767  177736                  JSR     %7,TYPD         ;GO TYPE CHAR,
003202  000734                          BR      TYPA
003204  000000                  TYPDAT: OPEN
                                        ;
                                        ;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
003206  011600                  TYPS:   MOV     @%6,%0          ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
003210  062716  000002                  ADD     #2,@%6          ;UPDATE TO NEXT MESSAGE ADDRESS
003214  011067  000014                  MOV     @%0,TYPSB       ;ADDRESS OF MESSAGE TO TYPSB
003220  022767  177777  000006          CMP     #-1,TYPSB       ;CHECK FOR TERMINATOR
003226  001001                          BNE     TYPSA           ;BRANCH IF NOT TERMINATOR,
003230  000002                          RTI                     ;TERMINATOR, EXIT
003232  104000                  TYPSA:  TYPE                    ;CALL ON TYP SUB TO TYPE MESSAGE
003234  000000                  TYPSB:  OPEN                    ;ADDRESS OF MESSAGE GOES HERE
003236  000763                          BR      TYPS            ;GO PROCESS NEXT MESSAGE
                                        ;
                                        ;OVERLAY VECTOR AREA
003240  012701  000300          OVRLAY: MOV     #300,%1         ;GET DL11-E VECTOR BASE ADDRESS
003244  012702  000302                  MOV     #302,%2
003250  012703  000004                  MOV     #4,%3
003254  010221                  OVRLYA: MOV     %2,(%1)+        ;LOAD VECTOR WITH IOT ERROR TRAP
003256  010321                          MOV     %3,(%1)+
003260  062702  000004                  ADD     #4,%2
003264  020127  001000                  CMP     %1,#1000                ;ALL VECTORS BEEN LOADED
003270  001401                          BEQ     OVRLYB
003272  000770                          BR      OVRLYA
003274  000207                  OVRLYB: RTS     7               ;EXIT
```

```
                                        ;SUBROUTINE TO DELAY A SPECIFIED NUMBER OF MILLISECONDS
003276  011667  000040          DLY:    MOV     @%6,DLCNT       ;GET DELAY COUNT ADDRESS,
003302  062716  000002                  ADD     #2,%6           ;SET UP EXIT ADDRESS
003306  017746  000030                  MOV     @DLCNT,-(6)     ;DELAY COUNT TO STACK
003312  001411                          BEQ     DLYC
003314  005067  174456                  CLR     PSW             ;SET PRIORITY 0
003320  012746  000226          DLYA:   MOV     #226,-(6)       ;1 MSEC COUNT TO STACK
003324  005316                  DLYB:   DEC     @%6             ;DECREMENT 1 MSEC COUNT
003326  001376                          BNE     DLYB            ;BRANCH IF NOT 0,
003330  005726                          POPSP                   ;ZERO, UNLOVER MSECS, COUNT,
003332  005316                          DEC     @%6             ;DECREMENT IT
003334  001371                          BNE     DLYA            ;BR IF NOT DONE DELAYING
003336  005726                  DLYC:   POPSP                   ;DONE
003340  000002                          RTI                     ;EXIT,
003342  000000                  DLCNT:  .OPEN                   ;CONTAINS MILLISECONDS COUNT ADDRESS,
                                        ;
                                        ;SUBROUTINE TO STALL A RANDOM NUMBER OF MILLISECONDS, MAXIMUM STALL
                                        ;DETERMINED BY CONTENTS OF LOC STLMSK,
003344  004767  177400          STAL:   JSR     %7,RNGEN        ;GO GET RANDOM NUMBER,
003350  046700  176026                  BIC     STLMSK,%0       ;# IN R0, APPLY STALL MASK,
003354  001404                          BEQ     STALB           ;BRANCH IF RESULT IS 0,
003356  010067  000002                  MOV     %0,STALA
003362  104016                          DELAY                   ;DELAY
003364  000000                  STALA:  .OPEN                   ;DELAY COUNT
003366  000002                  STALB:  RTI                     ;DONE, EXIT,
                                        ;
                                        ;SUBROUTINE TO GENERATE RANDOM CHARACTER COUNT
003370  004767  177354          GRCNT:  JSR     %7,RNGEN        ;GET RANDOM NUMBER
003374  046700  000010                  BIC     RCMSK,%0        ;APPLY MASK
003400  001773                          BEQ     GRCNT           ;TRY AGAIN IF RESULT 0
003402  010067  000004                  MOV     %0,RNCNT        ;COUNT TO RNCNT
003406  000207                          RTS     %7              ;EXIT,
003410  000000                  RCMSK:  .OPEN                   ;RANDOM CHARACTER MASK,
003412  000000                  RNCNT:  .OPEN                   ;RANDOM CHARACTER COUNT,
                                        ;
                                        ;SUBROUTINE TO SKIP ON FLAG AND TIME OUT IF SKIP FAILS
003414  016767  175764  000040  TMRX:   MOV     RXCSR,SIOT      ;SET UP RXCSR ADDRESS
003422  000403                          BR      TIME1
003424  016767  175760  000030  TMTX:   MOV     TXCSR,SIOT      ;SET UP TXCSR ADDRESS
003432  005067  000022          TIME1:  CLR     TIMER
003436  005267  000016          TIME2:  INC     TIMER
003442  001405                          BEQ     TIMEX           ;BRANCH IF COUNTER OVERFLOW
003444  105777  000012                  TSTB    @SIOT
003450  100372                          BPL     TIME2
003452  062716  000002                  ADD     #2,@%6          ;SET UP EXIT RETURN
003456  000002                  TIMEX:  RTI
003460  000000                  TIMER:  0
003462  000000                  SIOT:   0
```

```
                                        ;SUBROUTINE TO SELECT LINE
003464  032767  010000  174076  LINSEL: BIT     #BIT12,SR
003472  001003                          BNE     LINSLX          ;BRANCH IF SET
003474  005067  176106                  CLR     @QUNOV
003500  000205                          RTS     5
003502  004767  177532          LINSLX: JSR     %7,OVRLAY
003506  004767  177310                  JSR     %7,CLRCQ
003512  104000                          TYPE
003514  015541                          LDLINE
003516  000000                          HALT
003520  016767  174044  176064          MOV     SR,TEMP
003526  042767  177740  176056          BIC     #177740,TEMP
003534  016767  176052  176046          MOV     TEMP,LINENO     ;SAVE FOR TYPING
003542  006367  176044                  ASL     TEMP
003546  016701  176040                  MOV     TEMP,%1
003552  016101  001200                  MOV     RXCR0(1),%1     ;GET RXCSR DEVICE ADDRESS
003556  032701  000001                  BIT     #BIT0,%1        ;IS DEVICE THERE
003562  001403                          BEQ     LINB            ;YES
003564  104000          LINA:   TYPE                            ;NO, REPORT
003566  016100                          MNOLIN
003570  000744                          BR      LINSLX
003572  004767  001644          LINB:   JSR     %7,FORMAD
003576  005067  174174                  CLR     PSW
003602  052767  000001  175656          BIS     #BIT0,FMAP      ;SET MAPPING FLAG
003610  042777  000100  175572          BIC     #BIT6,@TXCSR
003616  052777  000100  175564          BIS     #BIT6,@TXCSR
003624  000240                          NOP
003626  000240                          NOP
003630  005767  175560                  TST     RXVTR
003634  001753                          BEQ     LINA
003636  042777  000100  175544          BIC     #BIT6,@TXCSR
003644  012767  000340  174124          MOV     #PRTY7,PSW
003652  004567  000166                  JSR     5,OACNV         ;TYPE LINE #
003656  001610                          LINENO
003660  015606                          SELINE
003662  000002                          2
003664  104000                          TYPE
003666  015574                          ALINE
003670  000205                          RTS     5
```

```
                                        ;
                                        ;SUBROUTINE TO INITIALIZE BINARY COUNT PATTERNS
003672  012767  177777  000014  INRINI: MOV     #-1,RIND          ;SET ALL VARIABLES
003700  004567  000226                  JSR     %5,RMOVE          ;TO MINUS 1.
003704  003714                          RIND
003706  003715                          RIND+1
003710  000013                          11.
003712  000207                          RTS     %7                ;EXIT
003714  000000          RIND:   OPEN
003716  000000          PT3:    OPEN
003720  000000          PT1:    OPEN
003722  000000          RINDP:  OPEN
003724  000000          PT0P:   OPEN
003726  000000          PT1P:   OPEN
                                        ;
                                        ;SPECIAL BINARY COUNT PATTERN SUBROUTINE, EXITS WITH BIN CHAR IN R0
003730  016767  177762  177762  GTBIN:  MOV     PT0,PT1           ;PREVIOUS BIN CHAR TO PT1
003736  005167  177756                  COM     PT1
003742  005167  177746                  COM     RIND
003746  001002                          BNE     .+6
003750  005267  177744                  INC     PT1
003754  042767  177400  177736        BIC     #177400,PT1       ;MASK TO 8 BITS
003762  016767  177732  177726        MOV     PT1,PT0           ;SAVE BIN CHAR IN PT0
003770  016700  177724                  MOV     PT1,%0            ;BIN CHAR TO R0,
003774  000207                          RTS     %7                ;EXIT,
003776  016767  177722  177722  GTBINP: MOV     PT0P,PT1P         ;PREVIOUS BIN CHAR TO PT1P
004004  005167  177716                  COM     PT1P
004010  005167  177706                  COM     RIND
004014  001002                          BNE     .+6
004016  005267  177704                  INC     PT1P
004022  042767  177400  177676        BIC     #177400,PT1P      ;MASK TO 8 BITS,
004030  016767  177672  177666        MOV     PT1P,PT0P         ;SAVE BIN CHAR IN PT0P,
004036  016701  177664                  MOV     PT1P,%1           ;BIN CHAR TO R1,
004042  000207                          RTS     %7                ;EXIT,
                                        ;
                                        ;OCTAL TO ASCII CONVERT ROUTINE
004044  104013          OACNV:  SAVREG
004046  013567  000056                  MOV     @(3)+,OACNVX      ;GET OCTAL VALUE,
004052  012501                          MOV     (3)+,%1           ;GET DESTINATION ADDR,
004054  012502                          MOV     (5)+,%2           ;GET CONVERT COUNT,
004056  060201                          ADD     %2,%1             ;DEVELOP ADDR TO STORE 1ST CHAR,
004060  016703  000044          OACNVA: MOV     OACNVX,%3
004064  042703  177770                  BIC     #177770,%3        ;ISOLATE LEAST SIGNIFICANT DIGIT,
004070  062703  000060                  ADD     #60,%3            ;CONVERT DIGIT TO ASCII,
004074  110341                          MOVB    %3,-(1)           ;STORE ASCII CHARACTER,
004076  042767  000007  000024        BIC     #7,OACNVX
004104  006067  000020                  ROR     OACNVX
004110  006067  000014                  ROR     OACNVX
004114  006067  000010                  ROR     OACNVX
004120  005302                          DEC     %2                ;DONE ALL DIGITS?
004122  001356                          BNE     OACNVA            ;BRANCH IF NOT DONE,
004124  104014                          RSTREG
004126  000205                          RTS     %5                ;DONE, EXIT,
004130  000000          OACNVX: OPEN
```

```
                                        ;SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES,
004132  104013                  BMOVE:  SAVREG                  ;SAVE REGS,
004134  012501                          MOV     (5)+,%1         ;GET "FROM" ADDRESS
004136  012502                          MOV     (5)+,%2         ;GET "TO" ADDRESS
004140  012503                          MOV     (5)+,%3         ;GET COUNT
004142  112122                  BMOVA:  MOVB    (1)+,(2)+       ;MOVE BYTE
004144  005303                          DEC     %3              ;DECREMENT COUNT
004146  001375                          BNE     BMOVA           ;BRANCH IF NOT DONE,
004150  104014                          RSTREG                  ;RESTORE REGS,
004152  000205                          RTS     %5              ;DONE EXIT
                                        ;
                                        ;BINARY TO DECIMAL ASCII CONVERT SUBROUTINE,
004154  104013                  BDCNV:  SAVREG
004156  012700  004332                  MOV     #DECVAL,%0      ;SET UP ADDR TO STORE DECIMAL ASCII IN R0
004162  013501                          MOV     @(5)+,%1        ;BINARY VALUE TO R1,
004164  012567  000052                  MOV     (5)+,BDCNVC     ;GET DEST ADDR
004170  012567  000050                  MOV     (5)+,BDCNVB     ;GET CHAR COUNT
004174  012702  004320                  MOV     #ADTENP,%2      ;ADDR OF TEN POWER STRING TO R2,
004200  012767  000005  000104          MOV     #5,CNVCTR       ;SET UP FOR 5 POWER CONVERSIONS,
004206  012267  000104          BDCNVA: MOV     (2)+,TENPWR     ;MOVE POWER OF TEN VALUE TO TENPWR,
004212  004767  000034                  JSR     %7,SUBTEN       ;PERFORM CONVERSION
004216  005367  000070                  DEC     CNVCTR          ;DONE 5 CONVERSIONS?
004222  001371                          BNE     BDCNVA          ;BRANCH IF NOT YET 5,
004224  166700  000014                  SUB     BDCNVD,%0
004230  010067  000004                  MOV     %0,BDCNVB
004234  004567  177672                  JSR     %5,BMOVE
004240  000000          BDCNVB: 0
004242  000000          BDCNVC: 0
004244  000000          BDCNVD: 0
004246  104014                          RSTREG
004250  000205                          RTS     %5              ;YES, EXIT,
004252  005067  000036          SUBTEN: CLR     DIGIT           ;CLEAR DIGIT
004256  166701  000034          SUBTNA: SUB     TENPWR,%1       ;SUBTRACT TEN POWER FROM BINARY VALUE,
004262  103403                          BCS     SUBTNB          ;BRANCH IF UNSUCCESSFUL SUBTRACTION,
004264  005267  000024                  INC     DIGIT
004270  000772                          BR      SUBTNA
004272  066701  000020          SUBTNB: ADD     TENPWR,%1       ;RESTORE SUBTRACTED VALUE,
004276  062767  000060  000010          ADD     #60,DIGIT       ;CONVERT (DIGIT) TO ASCII
004304  116720  000004                  MOVB    DIGIT,(0)+      ;MOVE ASCII CHAR TO DECVAL FIELD,
004310  000207                          RTS     %7              ;EXIT,
004312  000000          CNVCTR: OPEN
004314  000000          DIGIT:  OPEN
004316  000000          TENPWR: OPEN
004320  023420          ADTENP: 10000,
004322  001750                  1000,
004324  000144                  100,
004326  000012                  10,
004330  000001                  1
004332    040     040     040   DECVAL: .BYTE   040,040,040,040,040,040
004335    040     040     040
```

```
004340  742777  000002  175036  DATTST: BIC     #BIT1,@RXCSR    ;CLEAR DATA TERM, READY
004346  052777  000004  175034          BIS     #BIT2,@TXCSR    ;SET MAINTENANCE BIT
004354  012767  000144  175176          MOV     #100.,CTR0      ;GET CHARACTER COUNT
004362  105777  175022          DATAA:  TSTB    @TXCSR          ;WAIT FOR
004366  100375                          BPL     .-4             ;READY BIAS
004370  004767  177402                  JSR     7,GTBINP        ;GET CHARACTER
004374  110167  175154                  MOVB    %1,CRBUFA       ;MOVE CHARACTER
004400  004767  000340                  JSR     7,MASKIT        ;MASK OFF NON TRANSMITTED BITS
004404  110177  175072                  MOVB    %1,@TXBUF       ;TRANSMIT CHARACTER
004410  105777  174770                  TSTB    @RXCSR          ;WAIT FOR
004414  100375                          BPL     .-4             ;DONE FLAG
004416  117767  174764  175126          MOVB    @RXBUF,CRBUF    ;GET RECEIVED CHARACTER
004424  104004                          DATCHK                  ;CHK DATA
004426  005367  175126                  DEC     CTR0            ;DECREMENT CHARACTER COUNT
004432  001353                          BNE     DATAA
004434  005726                          TST     (6)+            ;POP STACK
004436  104012                          SCOPE
                                ;
004440  104000                  SETSR:  TYPE                    ;TYPE SELECT OPTION MESSAGE,
004442  015000                          ASETSR
004444  000000                          HALT                    ;COMMON HALT,
004446  000207                          RTS     %7              ;EXIT;
004450  104000                  INCRTN: TYPE                    ;TYPE INCORRECT ROUTINE SELECTED,
004452  015076                          AINCRT
004454  000000                          HALT                    ;COMMON HALT,
004456  000207                          RTS     %7              ;EXIT;
004460  104000                  INCRPG: TYPE
004462  015217                          AINCPG
004464  000000                          HALT
004466  000167  175142                  JMP     START
004472  005067  175110          PRGEND: CLR     FOUNDV
004476  032767  020000  173064          BIT     #BIT13,SR       ;INHIBIT PRINT SET?
004504  001026                          BNE     PRGEXT          ;BR IF SET
004506  004567  177442                  JSR     %5,BDCNV
004512  001626                          PASCNT
004514  015267                          APCNT
004516  000006                          6
004520  004567  177420                  JSR     %5,DACNV        ;CONVERT LINE NUMBER
004524  001610                          LINENO
004526  015307                          ACLIN
004530  000002                          2
004532  004567  177306                  JSR     %5,DACNV        ;CONVERT RXCSR
004536  001404                          RXCSR
004540  015323                          APRXC
004542  000006                          6
```

```
004544  004567  177274          JSR     %5,DACNV        ;CONVERT VECTOR
004550  001414                  RXVTR
004552  015344                  APVEC
004554  000004                  4
004556  104000                  TYPE                    ;TYPE PROGRAM END',
004560  015252                  APGEND
004562  032767  010000  173000  PRGEXT: BIT     #BIT12,SR       ;LOOK ON LINE
004570  001403                  BEQ     PRGXT1          ;BR IF NOT SET
004572  005267  175030          INC     PASCNT
004576  000425                  BR      PRGXTL
004600  016767  175004  175004  PRGXT1: MOV     LINENO,TEMP     ;GET LINENO
004606  006367  175000          ASL     TEMP
004612  062767  000002  174772  PRGEC: ADD     #2,TEMP         ;UPDATE LINE NUMBER
004620  016701  174766          PRGEA: MOV     TEMP,%1
004624  016101  001200          MOV     RXCSR0(1),%1    ;GET RXCSR DEVICE ADDRESS
004630  022701  177777          CMP     #177777,%1      ;LAST ONE
004634  001023                  BNE     PRGEB           ;NO,CONTINUE
004636  005267  174764          INC     PASCNT
004642  005067  174742          CLR     LINENO
004646  005067  174740          CLR     TEMP
004652  013705  000042  PRGXTL: MOV     @#42,%5
004656  001405                  BEQ     CONT
004660  000005                  RESET
004662  004715          LOGIC:  JSR     7,(5)
004664  000240                  NOP
004666  000240                  NOP
004670  000240                  NOP
004672  032767  010000  172670  CONT:   BIT     #BIT12,SR       ;LOCK ON LINE
004700  001747                  BEQ     PRGEA           ;BRANCH IF NOT SET
004702  000207                  RTS     7
004704  032701  000001  PRGEB:  BIT     #BIT0,%1        ;DEVICE THERE
004710  001340                  BNE     PRGEC           ;NO
004712  006267  174674          ASR     TEMP
004716  016767  174670  174664  MOV     TEMP,LINENO
004724  004767  000512          JSR     %7,FORMAD
004730  000207                  RTS     %7              ;EXIT',
                        ;
                        ;CONDITIONAL ERROR HALT ROUTINE.
004732  005767  172632  EHLT:   TST     SR              ;CHECK FOR HALT ON ERROR',
004736  100001                  BPL     EHLTA           ;BRANCH IF NO HALT DESIRED',
004740  000000                  HALT                    ;HALT',
004742  000002          EHLTA:  RTI                     ;IN DATA LIGHTS,
                        ;
                        ;MASKIT = MASK DATA ACCORDING TO LINE NUMBER
004744  016767  174426  174426  MASKIT: MOV     UMASK,RMASK     ;GET MASK
004752  042767  177000  174420          BIC     #177000,RMASK   ;REMOVE C/D FLAG+PRIORITY
004760  005167  174414                  COM     RMASK
004764  046767  174410  174562          BIC     RMASK,CRBUFA    ;MASK DESIRED BITS
004772  000207                  RTS     7
```

```
                                        ;
                                        ;DATA CHECK ROUTINE, TEST ERROR BITS
004774  017767  174436  174554  DTCHK1  MOV     @RXBUF,CRBUFB   ;DID ANY ERROR BITS SET
005002  032767  170000  174546          BIT     #170000,CRBUFB
005010  001004                          BNE     DTCHKX          ;YES, TYPE ERROR
005012  026767  174534  174536          CMP     CRBUF,CRBUFA    ;COMPARE EXPECTED AND RECEIVED
005020  001421                          BEQ     DTCHKA          ;CHARS, BRANCH IF SAME;
005022  004567  177016          DTCHKX: JSR     %5,OACNV        ;GO TO OCTAL TO ASCII CONVERT;
005026  001552                          CRBUF                   ;SOURCE ADDR;
005030  014755                          AWAS                    ;DESTINATION ADDR;
005032  000003                          3                       ;#OF DIGITS TO CONVERT;
005034  004567  177004                  JSR     %5,OACNV        ;GO TO OCTAL TO ASCII CONVERT;
005040  001554                          CRBUFA                  ;SOURCE ADDR;
005042  014744                          ARCB                    ;DESTINATION ADDR;
005044  000003                          3                       ;#OF DIGITS TO CONVERT;
005046  004567  176772                  JSR     %5,OACNV
005052  001556                          CRBUFB
005054  014770                          ARXBUF
005056  000006                          6
005060  104015                          ERROR1
005062  014732                          ERRAT
005064  000002          DTCHKA: RTI
                                        ;
                                        ;ERROR HANDLER
005066  012767  177777  000142  ERR:    MOV     #-1,ERRB        ;SET UP ONE MESSAGE CALL;
005074  012767  000240  000136          MOV     #240,ERRB+2
005102  005067  000146                  CLR     ERRE
005106  000413                          BR      ERRA
005110  011667  000122          ERR1:   MOV     @%6,ERRB        ;DEVELOP ADDTNL MESSAGE ADDR;
005114  017767  000116  000114          MOV     @ERRB,ERRB      ;STORE AT ERRB;
005122  012767  177777  000110          MOV     #-1,ERRB+2
005130  012767  000002  000116          MOV     #2,ERRE
005136  032767  020000  172424  ERRA:   BIT     #BIT13,SR       ;INHIBIT ERROR PRINT?
005144  001036                          BNE     ERRC            ;BRANCH TO INHIBIT PRINT;
005146  011667  000100                  MOV     @%4,ERRD        ;DEVELOP CALLING ADDR;
005152  162767  000002  000072          SUB     #2,ERRD
005160  016767  174266  174266          MOV     RTINO,TNNO
005166  042767  100000  174260          BIC     #BIT15,TNNO
005174  004567  176644                  JSR     %5,OACNV        ;GO TO OCTAL TO ASCII CONVERT;
005200  005252                          ERRD                    ;SOURCE ADDR;
005202  014147                          APC                     ;DESTINATION ADDR;
005204  000006                          6                       ;#OF DIGITS TO CONVERT;
005206  004567  176632                  JSR     %5,OACNV        ;GO TO OCTAL TO ASCII CONVERT;
005212  001404                          RXCSR                   ;SOURCE ADDR;
005214  014166                          RRXNUM                  ;DESTINATION ADDR;
005216  000006                          6                       ;#OF DIGITS TO CONVERT;
005220  004567  176620                  JSR     %5,OACNV        ;GO TO OCTAL TO ASCII CONVERT;
005224  001452                          RTNNO                   ;SOURCE ADDR;
005226  014137                          ATNUMB                  ;DESTINATION ADDR;
005230  000003                          3                       ;#OF DIGITS TO CONVERT;
005232  104001                          TYPES                   ;TYPE;
005234  014135                          EMG                     ;ERROR HEADER;
```

```
005236  000000                  ERRBI   OPEN                    ;ADDTNL ERROR MESSAGE IF ANY.
005240  177777                          -1
005242  104010                  ERRCI   EHALT                   ;GO ERR HALT IF DESIRED.
005244  066716  000004                  ADD     ERRE,@%6
005250  000002                          RTI                     ;EXIT.
005252  000000                  ERRDI   OPEN
005254  000000                  ERREI   OPEN
                                ;
                                ;ERROR TRAP HANDLER - TYPE TO AND FROM WHERE ERROR TRAP OCCURRED
005256  016767  172514  174200  ERTPI   MOV     PSW,OLDPS       ;SAVE OLD STATUS
005264  012767  000340  172504          MOV     #PRTY7,PSW
005272  006267  174166                  ASR     OLDPS
005276  006267  174162                  ASR     OLDPS
005302  006267  174156                  ASR     OLDPS
005306  042767  177740  174150          BIC     #177740,OLDPS
005314  016767  174144  174300          MOV     OLDPS,TOPC
005322  011667  174270                  MOV     @%6,FROMPC      ;GET FROM PC
005326  004567  176512          ERTPAI  JSR     %5,DACNV
005332  001622                          TOPC
005334  016035                          MTO
005336  000006                          6
005340  004567  176500                  JSR     %5,DACNV
005344  001624                          FROMPC
005346  016067                          MFROM
005350  000006                          6
005352  104000                          TYPE
005354  015770                          MTERR
005356  000000                          HALT
005360  000167  174250                  JMP     START
                                ;
                                ;MAPVEC - MAP VECTOR OR REPORT ERROR DEPENDING ON FMAP FLAG
005364  011667  174232          MAPVECI MOV     @%6,TOPC
005370  022626                          POPSP2
005372  011667  174226                  MOV     @%6,FROMPC
005376  162767  000004  174216          SUB     #4,TOPC
005404  005767  174056                  TST     FMAP
005410  001746                          BEQ     ERTPA           ;NOT MAPPING, REPORT ERROR
005412  016767  174204  174000          MOV     TOPC,TXVTR      ;STORE VECTOR
005420  162767  000004  174174          SUB     #4,TOPC
005426  016767  174170  173760          MOV     TOPC,RXVTR
005434  005067  174026                  CLR     FMAP
005440  000002                          RTI
```

```
                                        ;
                                        ;FORMAD=FORM DEVICE AT ADDRESSES
005442  010167  173736                  FORMAD: MOV     %1,RXCSR
005446  062701  000002                          ADD     #2,%1
005452  010167  173730                          MOV     %1,RXBUF
005456  062701  000002                          ADD     #2,%1
005462  010167  173722                          MOV     %1,TXCSR
005466  062701  000002                          ADD     #2,%1
005472  010167  173714                          MOV     %1,TXBUF
005476  016767  174106  174106                  MOV     LINENO,TEMP     ;GET PRIORITY
005504  006367  174102                          ASL     TEMP
005510  062767  001300  174074                  ADD     #CMASK,TEMP
005516  017767  174070  174070                  MOV     @TEMP,TEMP1
005524  016767  174064  173644                  MOV     TEMP1,UMASK
005532  000367  174056                          SWAB    TEMP1
005536  006367  174052                          ASL     TEMP1
005542  042767  177437  174044                  BIC     #177437,TEMP1
005550  016767  174040  173640                  MOV     TEMP1,RXLVL
005556  016767  174032  173636                  MOV     TEMP1,TXLVL
005564  000207                                  RTS     %7
                                        ;
                                        ;DOTHIS = SELECTABLE TEST DECISION MAKER
                                        ;
005566  032767  001000  171774        DOTHIS: BIT     #BIT9,SR        ;IS SELECT TEST SWITCH SET
005574  001002                                BNE     GOBACK          ;RETURN TO TEST IF SW SET
005576  000167  174370                        JMP     GTRQYX          ;GO TO NEXT TEST
005602  000207                        GOBACK: RTS     %7
                                        ;
005604  012767  005614  172212        PFAIL:  MOV     #PWRUP,24
005612  000000                                HALT
005614  012767  005604  172202        PWRUP:  MOV     #PFAIL,24
005622  000005                                RESET
005624  012706  001176                        MOV     #SPBOT,%6
005630  104000                                TYPE
005632  016243                                .MPWRP
005634  104003                                ERROR
005636  000445                                BR      RESTART
```

```
                                        ;
                                        ;DECIDE IF VECTOR TO BE MAPPED AND MAP
005640  022767  000000  173740  TSTVEC: CMP     #2,FOUNDV       ;NEED VECTOR MAPPING
005646  001040                          BNE     TSTVEX          ;NO, EXIT
005650  004767  175364                  JSR     %7,OVRLAY
005654  005067  173534                  CLR     RXVTR
005660  005067  172112                  CLR     PSW
005664  052767  000001  173574          BIS     #BIT0,FMAP      ;SET MAPPING FLAG
005672  042777  000100  173510          BIC     #BIT6,@TXCSR    ;CAUSE INTERRUPT
005700  052777  000100  173502          BIS     #BIT6,@TXCSR
005706  000240                          NOP
005710  000240                          NOP
005712  005767  173476                  TST     RXVTR           ;DID TRAP OCCUR?
005716  001004                          BNE     TSTVA           ;YES, OK
005720  104020                          TYPE                    ;NO, ERROR
005722  016100                          MNOLINE
005724  000167  173704                  JMP     START
005730  042777  000100  173452  TSTVA:  BIC     #BIT6,@TXCSR
005736  012767  000340  172032          MOV     #PRTY7,PSW      ;RAISE PRIORITY, RETURN
005744  005267  173036                  INC     FOUNDV
005750  000207          .       TSTVEX: RTS     %7
                                        ;
                                        ;RESTART ROUTINE
005752  016700  173466          RESTART:MOV     PRGNUM,%0
005756  006300                          ASL     %0
005760  000170  005764                  JMP     @RSTART(0)      ;GO RESTART SELECTED PROGRAM

005764  006036                  RSTART: PRG0A                   ;PROGRAM 0 RESTART ADDRESS
005766  013522                          PRG1A                   ;PROGRAM 1 RESTART ADDRESS
005770  013562                          PRG2A                   ;PROGRAM 2 RESTART ADDRESS
005772  013646                          PRG3A                   ;PROGRAM 3 RESTART ADDRESS
005774  013676                          PRG4A                   ;PROGRAM 4 RESTART ADDRESS
005776  004460                          INCRPG
006000  004460                          INCRPG
006002  004460                          INCRPG
```

```
                                        ;
                                        ;PROG 0 - INPUT-OUTPUT LOGIC TESTS
                                        ;
006004  012767  006042  173434  PRG0:   MOV     #AT0,KSTART
006012  005737  000042                  TST     @#42                    ;MONITOR BOOT
006016  001005                          BNE     PRG0B                   ;YES, START TEST
006020  104000                          TYPE                            ;TYPE TITLE AND INSTRUCTIONS
006022  014177                          .RPTIT
006024  000000                          HALT
006026  004767  176406                  JSR     7,SETSR
006032  004567  175426          PRG0B:  JSR     5,LINSEL                ;GO GET LINE # FROM USER
006036  000167  174122          PRG0A:  JMP     GETRDY                  ;GET STARTED,
        177777                          X=-1
                                        .TSTA   1000,,AA,CD
                                        .TSTAA  AA,1000,,\X+1+CD,\X+2,\X+1
                                        ;*****************************************************************************
006042  100000                  AT0:    100000                          ;TEST NUMBER                                *
006044  006074                          AT1                             ;ADDRESS OF NEXT TEST                       *
006046  001750                          1000,                           ;ITERATION COUNT                            *
006050  006052                          AAA                             ;SCOPE ENTRY POINT                          *
        000000                          X=X+1                           ;                                           *
                                        ;*****************************************************************************
                                        ;TEST ABILITY TO REFERENCE RECEIVER CSR WITHOUT TRAPPING
006052  012767  006066  171724  AAA:    MOV     #AAE,@#ACHER            ;SET UP MACHINE ERROR TRAP,
006060  005777  173320                  TST     @#RXCSR                 ;REFERENCE RXCSR
006064  104012                  AAB:    SCOPE                           ;OK IF NO TRAP, SCOPE
006066  022626                  AAE:    CMP(SP)+,(SP)+
006070  104003                          ERROR                           ;TRAPPED WHEN REFERENCING RXCSR,
006072  000774                          BR      AAB
                                        .TSTA   1000,,AB,CD
                                        .TSTAA  AB,1000,,\X+1+CD,\X+2,\X+1
                                        ;*****************************************************************************
006074  100001                  AT1:    100001                          ;TEST NUMBER                                *
006076  006126                          AT2                             ;ADDRESS OF NEXT TEST                       *
006100  001750                          1000,                           ;ITERATION COUNT                            *
006102  006104                          ABA                             ;SCOPE ENTRY POINT                          *
        000001                          X=X+1                           ;                                           *
                                        ;*****************************************************************************
                                        ;TEST ABILITY TO REFERENCE RECEIVER BUFFER WITHOUT TRAPPING
006104  012767  006120  171672  ABA:    MOV     #ABE,@#ACHER            ;SET UP MACHINE ERROR TRAP,
006112  005777  173270                  TST     @#RXBUF                 ;REFERENCE RXBUF
006116  104012                  ABB:    SCOPE                           ;OK IF NO TRAP SCOPE
006120  022626                  ABE:    CMP(SP)+,(SP)+
006122  104003                          ERROR                           ;TRAPPED WHEN REFERENCING RXBUF
006124  000774                          BR      ABB
                                        .TSTA   1000,,AC,CD
                                        .TSTAA  AC,1000,,\X+1+CD,\X+2,\X+1
                                        ;*****************************************************************************
006126  100002                  AT2:    100002                          ;TEST NUMBER                                *
006130  006160                          AT3                             ;ADDRESS OF NEXT TEST                       *
006132  001750                          1000,                           ;ITERATION COUNT                            *
006134  006136                          ACA                             ;SCOPE ENTRY POINT                          *
        000002                          X=X+1                           ;                                           *
                                        ;*****************************************************************************
```

```
                                ;TEST ABILITY TO REFERENCE TRANSMITTER CSR WITHOUT TRAPPING,
006136  012767  006152  171640  ACA:    MOV     #ACE,MACHER             ;SET UP MACHINE ERROR TRAP',
006144  005777  173240          ACB:    TST     @TXCSR                  ;REFERENCE TXCSR
006150  104012                  ACB:    SCOPE                           ;SCOPE
006152  022626                  ACE:    POPSP2
006154  104003                          ERROR                           ;TRAPPED WHEN REFERENCING TXCSR
006156  000774                          BR      ACB
                                        TSTA    1000,,AD,CD
                                        TSTAA   AD,1000,,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************
006160  100003                  AT3:    100003                          ;TEST NUMBER                *
006162  006212                          AT4                             ;ADDRESS OF NEXT TEST       *
006164  001750                          1000.                           ;ITERATION COUNT            *
006166  006170                          ADA                             ;SCOPE ENTRY POINT          *
        000003                          X=X+1                           ;                           *
                                ;*****************************************************************
                                ;TEST ABILITY TO REFERENCE TRANSMITTER BUFFER WITHOUT TRAPPING
006170  012767  006204  171606  ADA:    MOV     #ADE,MACHER             ;SET UP MACHINE ERROR TRAP',
006176  005777  173210          ADB:    TST     @TXBUF                  ;REFERENCE TX BUF',
006202  104012                  ADB:    SCOPE                           ;SCOPE
006204  022626                  ADE:    POPSP2
006206  104003                          ERROR                           ;TRAPPED WHEN REFERENCING TXBUF
006210  000774                          BR      ADB
                                        TSTA    10',,AE,CD
                                        TSTAA   AE,10',,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************
006212  100004                  AT4:    100004                          ;TEST NUMBER                *
006214  006312                          AT5                             ;ADDRESS OF NEXT TEST       *
006216  000012                          10.                             ;ITERATION COUNT            *
006220  006222                          AEA                             ;SCOPE ENTRY POINT          *
        000004                          X=X+1                           ;                           *
                                ;*****************************************************************
                                ;TEST THAT TXCSR BIT 0 (BREAK) CAN BE SET AND CLEARED
                                ;AND THAT RESET CLEARS IT
006222  032777  000001  173160  AEA:    BIT     #BIT0,@TXCSR            ;SEE IF BIT IS CLEAR
006230  001402                          BEQ     AEB                     ;BR IF CLEAR
006232  104003                          ERROR                           ;RESET DID NOT CLEAR IT
006234  000421                          BR      AED
006236  052777  000001  173144  AEB:    BIS     #BIT0,@TXCSR            ;SET TXCSR BIT 0
006244  032777  000001  173136          BIT     #BIT0,@TXCSR            ;DID IT SET
006252  001002                          BNE     AEC                     ;YES, GO ON
006254  104003                          ERROR                           ;TXCSR BIT0 FAILED TO SET
006256  000410                          BR      AED
006260  042777  000001  173122  AEC:    BIC     #BIT0,@TXCSR            ;CLEAR TXCSR BIT 0
006266  032777  000001  173114          BIT     #BIT0,@TXCSR            ;DID IT CLEAR
006274  001401                          BEQ     AED
006276  104003                          ERROR                           ;TXCSR BIT 0 DID NOT CLEAR
006300  052777  000001  173102  AED:    BIS     #BIT0,@TXCSR
006306  104011                          SRESET                          ;ISSUE RESET TO CLEAR
006310  104012                          SCOPE
                                        TSTA    10',,AG,CD
                                        TSTAA   AG,10',,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************
006312  100005                  AT5:    100005                          ;TEST NUMBER                *
006314  006412                          AT6                             ;ADDRESS OF NEXT TEST       *
```

```
006316  000012                          10,                     ;ITERATION COUNT
006320  006322                          AGA                     ;SCOPE ENTRY POINT
        000005                          .=.+1                   ;
                        ;*****************************************************************
                        ;TEST THAT TXCSR BIT2 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
006322  032777  000004  173040  AGA:    BIT     #BIT2,@TXCSR    ;SEE IF TXCSR BIT2 IS CLEAR,
006330  001402                          BEQ     AGB             ;BRANCH IF BIT IS CLEAR,
006332  104003                          ERROR                   ;RESET DID NOT CLEAR TXCSR BIT2
006334  000421                          BR      AGD
006336  052777  000004  173044  AGB:    BIS     #BIT2,@TXCSR    ;SET TXCSR BIT2,
006344  032777  000004  173036          BIT     #BIT2,@TXCSR    ;SEE IF BIT IS SET,
006352  001002                          BNE     AGC             ;BRANCH IF BIT IS SET,
006354  104003                          ERROR                   ;TXCSR BIT2 FAILED TO SET,
006356  000410                          BR      AGD
006360  042777  000004  173022  AGC:    BIC     #BIT2,@TXCSR    ;CLEAR TXCSR BIT2
006366  032777  000004  173014          BIT     #BIT2,@TXCSR    ;SEE IF BIT IS CLEAR,
006374  001401                          BEQ     AGD
006376  104003                          ERROR                   ;TXCSR BIT2 FAILED TO CLEAR,
006400  052777  000004  173002  AGD:    BIS     #BIT2,@TXCSR    ;SET TXCSR BIT2,
006406  104011                          CRESET                  ;ISSUE RESET TO CLEAR BIT,
006410  104012                          SCOPE                   ;SCOPE
                                        TSTA    10.,,AJ,CD
                                        TSTAA   AJ,10.,,\X+1-CD,\X+2,\X+1
                        ;*****************************************************************
006412  100006          AT6:  100006                            ;TEST NUMBER
006414  006520                          AT7                     ;ADDRESS OF NEXT TEST
006416  000012                          10,                     ;ITERATION COUNT
006420  006422                          AJA                     ;SCOPE ENTRY POINT
        000006                          .=.+1                   ;
                        ;*****************************************************************
                        ;TEST THAT TXCSR BIT6 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
006422  012767  000340  171346  AJA:    MOV     #PRTY7,PSW      ;SET PRIORITY 7,
006430  032777  000100  172752          BIT     #BIT6,@TXCSR    ;SEE IF TXCSR BIT6 IS CLEAR,
006436  001402                          BEQ     AJB             ;BRANCH IF BIT IS CLEAR,
006440  104003                          ERROR                   ;RESET DID NOT CLEAR TXCSR BIT6
006442  000421                          BR      AJD
006444  052777  000100  172736  AJB:    BIS     #BIT6,@TXCSR    ;SET TXCSR BIT6,
006452  032777  000100  172730          BIT     #BIT6,@TXCSR    ;SEE IF BIT IS SET,
006460  001002                          BNE     AJC             ;BRANCH IF BIT IS SET,
006462  104003                          ERROR                   ;TXCSR BIT6 FAILED TO SET,
006464  000410                          BR      AJD
006466  042777  000100  172714  AJC:    BIC     #BIT6,@TXCSR    ;CLEAR TXCSR BIT6
006474  032777  000100  172706          BIT     #BIT6,@TXCSR    ;SEE IF BIT IS CLEAR,
006502  001401                          BEQ     AJD
006504  104003                          ERROR                   ;TXCSR BIT6 FAILED TO CLEAR,
006506  052777  000100  172674  AJD:    BIS     #BIT6,@TXCSR    ;SET TXCSR BIT6,
006514  104011                          SRESET                  ;ISSUE RESET TO CLEAR BIT,
006516  104012                          SCOPE                   ;SCOPE
                                        TSTA    100,,AK,CD
                                        TSTAA   AK,100,,\X+1-CD,\X+2,\X+1
                        ;*****************************************************************
006520  100007          AT7:  100007                            ;TEST NUMBER
006522  006544                          AT10                    ;ADDRESS OF NEXT TEST
006524  000144                          100.                    ;ITERATION COUNT
006526  006530                          AKA                     ;SCOPE ENTRY POINT
```

```
                000007                          X=X+1                           ;
                                        ;*******************************************************************
                                        ;TEST THAT TXCSR BIT 7 (READY BIT) IS SET UPON ENTERING ROUTINE AND
                                        ;THAT IT CAN BE READ RELIABLY.
006530  105777  172654         AKA:    TSTB    @TXCSR                  ;SEE IF TXCSR BIT 7 IS SET;
006534  100402                         BMI     AKB                     ;BRANCH IF SET;
006536  104003                         ERROR                           ;TXCSR BIT 7 NOT SET;
006540  104011                         SRESET                          ;ISSUE RESET TO CLEAR BIT IF ERROR
006542  104012         AKB:    SCOPE                                   ;SCOPE
                                TSTA    100,,AL,0
                                TSTAA   AL,100,,\X+1+0,\X+2,\X+1
                                        ;*******************************************************************
006544  000010         AT10:   10                              ;TEST NUMBER                                    *
006546  006626                 AT11                            ;ADDRESS OF NEXT TEST                           *
006550  000144                 100.                            ;ITERATION COUNT                                *
006552  006554                 ALA                             ;SCOPE ENTRY POINT                              *
                000010          X=X+1                           ;                                               *
                                        ;*******************************************************************
                                        ;TEST THAT RXCSR BIT 1 CAN BE SET & CLEARED
006554  042777  000002  172622 ALA:    BIC     #BIT1,@RXCSR
006562  052777  000002  172614         BIS     #BIT1,@RXCSR            ;SET RXCSR BIT1
006570  032777  000002  172606         BIT     #BIT1,@RXCSR            ;SEE IF BIT IS SET
006576  001002                         BNE     ALY                     ;BRANCH IF SET
006600  104003                         ERROR                           ;RXCSR BIT 1 FAILED TO SET
006602  000410                         BR      ALZ
006604  042777  000002  172572 ALY:    BIC     #BIT1,@RXCSR            ;CLEAR RXCSR BIT 1
006612  032777  000002  172564         BIT     #BIT1,@RXCSR            ;SEE IF BIT IS CLEAR
006620  001401                         BEQ     ALZ
006622  104003                         ERROR                           ;RXCSR BIT 1 FAILED TO CLEAR
006624  104012         ALZ:    SCOPE                                   ;SCOPE
                                TSTA    10',,AP,0
                                TSTAA   AP,10',,\X+1+0,\X+2,\X+1
                                        ;*******************************************************************
006626  000011         AT11:   11                              ;TEST NUMBER                                    *
006630  006720                 AT12                            ;ADDRESS OF NEXT TEST                           *
006632  000012                 10.                             ;ITERATION COUNT                                *
006634  006636                 APA                             ;SCOPE ENTRY POINT                              *
                000011          X=X+1                           ;                                               *
                                        ;*******************************************************************
                                        ;TEST THAT RXCSR BIT2 IS CLEAR AND CAN BE READ RELIABLY.
006636  032777  000004  172540 APA:    BIT     #BIT2,@RXCSR            ;SEE IF RXCSR BIT2 IS CLEAR;
006644  001402                         BEQ     APB                     ;BRANCH IF BIT IS CLEAR;
006646  104003                         ERROR                           ;RXCSR BIT2 IS NOT CLEAR;
006650  000421                         BR      APD
006652  052777  000004  172524 APB:    BIS     #BIT2,@RXCSR            ;SET RXCSR BIT2
006660  032777  000004  172516         BIT     #BIT2,@RXCSR            ;SEE IF BIT IS SET
006666  001002                         BNE     APCX                    ;BRANCH IF SET
006670  104003                         ERROR                           ;RXCSR BIT2 FAILED TO SET
006672  000410                         BR      APD
006674  042777  000004  172502 APCX:   BIC     #BIT2,@RXCSR            ;CLEAR RXCSR BIT2
006702  032777  000004  172474         BIT     #BIT2,@RXCSR            ;SEE IF BIT IS CLEAR
006710  001401                         BEQ     APD
006712  104003                         ERROR                           ;RXCSR BIT2 FAILED TO CLEAR
006714  052777  000004  172462 APD:    BIS     #B[illegible]@RXCSR     ;SET BIT
006722  104011                         SRESET                          ;ISSUE RESET TO CLEAR BIT
```

```
006724  104012                          SCOPE
                                        TSTA    10',,AQ,0
                                        TSTAA   AQ,10',,\X+1+3,\X+2,\Y+1
                                ;*******************************************************************************
006726  000012                  AT12:   12                      ;TEST NUMBER                                    *
006730  007026                          AT13                    ;ADDRESS OF NEXT TEST                           *
006732  000012                          10';                    ;ITERATION COUNT                                *
006734  006736                          AQA                     ;SCOPE ENTRY POINT                              *
        000012                          X=X+1                   ;                                               *
                                ;*******************************************************************************
                                ;TEST THAT RXCSR BIT3 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT;
006736  032777  000010  172440  AQA:    BIT     #BIT3,@RXCSR    ;SEE IF RXCSR BIT3 IS CLEAR;
006744  001402                          BEQ     AQB             ;BRANCH IF BIT IS CLEAR;
006746  104003                          ERROR                   ;RESET DID NOT CLEAR RXCSR BIT3
006750  000421                          BR      AQD
006752  052777  000010  172424  AQB:    BIS     #BIT3,@RXCSR    ;SET RXCSR BIT3;
006760  032777  000010  172416          BIT     #BIT3,@RXCSR    ;SEE IF BIT IS SET;
006766  001002                          BNE     AQC             ;BRANCH IF BIT IS SET;
006770  104003                          ERROR                   ;RXCSR BIT3 FAILED TO SET;
006772  000410                          BR      AQD
006774  042777  000010  172402  AQC:    BIC     #BIT3,@RXCSR    ;CLEAR RXCSR BIT3
007002  032777  000010  172374          BIT     #BIT3,@RXCSR    ;SEE IF BIT IS CLEAR;
007010  001401                          BEQ     AQD
007012  104003                          ERROR                   ;RXCSR BIT3 FAILED TO CLEAR;
007014  052777  000010  172362  AQD:    BIS     #BIT3,@RXCSR    ;SET RXCSR BIT3;
007022  104011                          SRESET                  ;ISSUE RESET TO CLEAR BIT;
007024  104012                          SCOPE                   ;SCOPE
                                        TSTA    10',,AR,0
                                        TSTAA   AR,10',,\X+1+3,\X+2,\Y+1
                                ;*******************************************************************************
007026  000013                  AT13:   13                      ;TEST NUMBER                                    *
007030  007134                          AT14                    ;ADDRESS OF NEXT TEST                           *
007032  000012                          10';                    ;ITERATION COUNT                                *
007034  007036                          ARA                     ;SCOPE ENTRY POINT                              *
        000013                          X=X+1                   ;                                               *
                                ;*******************************************************************************
                                ;TEST THAT RXCSR BIT5 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT;
007036  012767  000340  170732  ARA:    MOV     #PRTY7,PSW      ;PRTY7 TO INHIBIT ANY INT
007044  032777  000040  172332          BIT     #BIT5,@RXCSR    ;SEE IF RXCSR BIT5 IS CLEAR;
007052  001402                          BEQ     ARB             ;BRANCH IF BIT IS CLEAR;
007054  104003                          ERROR                   ;RESET DID NOT CLEAR RXCSR BIT5
007056  000421                          BR      ARD
007060  052777  000040  172316  ARB:    BIS     #BIT5,@RXCSR    ;SET RXCSR BIT5;
007066  032777  000040  172310          BIT     #BIT5,@RXCSR    ;SEE IF BIT IS SET;
007074  001002                          BNE     ARC             ;BRANCH IF BIT IS SET;
007076  104003                          ERROR                   ;RXCSR BIT5 FAILED TO SET;
007100  000410                          BR      ARD
007102  042777  000040  172274  ARC:    BIC     #BIT5,@RXCSR    ;CLEAR RXCSR BIT5
007110  032777  000040  172266          BIT     #BIT5,@RXCSR    ;SEE IF BIT IS CLEAR;
007116  001401                          BEQ     ARD
007120  104003                          ERROR                   ;RXCSR BIT4 FAILED TO CLEAR;
007122  052777  000040  172254  ARD:    BIS     #BIT5,@RXCSR    ;SET RXCSR BIT5;
007130  104011                          SRESET                  ;ISSUE RESET TO CLEAR BIT;
007132  104012                          SCOPE                   ;SCOPE
                                        TSTA    10',,AS,0
```

```
                                        TSTAA   AS,10,,\X+1+CD,\Y+2,\X+1
                                ;*******************************************************************
007134  100014                  AT14:   100014                  ;TEST NUMBER                    *
007136  007242                          AT15                    ;ADDRESS OF NEXT TEST           *
007140  000012                          10.                     ;ITERATION COUNT                *
007142  007144                          ASA                     ;SCOPE ENTRY POINT              *
        000014                          X=X+1                   ;                               *
                                ;*******************************************************************
                                ;TEST THAT RXCSR BIT6 CAN BE SET, CLEARED, AND THAT RESET CLEARS IT.
007144  012767  000340  170624  ASA:    MOV     #PRTY7,PSW      ;SET PRIORITY 7,
007152  032777  000100  172224          BIT     #BIT6,@RXCSR    ;SEE IF RXCSR BIT6 IS CLEAR,
007160  001402                          BEQ     ASB             ;BRANCH IF BIT IS CLEAR,
007162  104003                          ERROR                   ;RESET DID NOT CLEAR RXCSR BIT6
007164  000421                          BR      ASD
007166  052777  000100  172210  ASB:    BIS     #BIT6,@RXCSR    ;SET RXCSR BIT6,
007174  032777  000100  172202          BIT     #BIT6,@RXCSR    ;SEE IF BIT IS SET,
007202  001002                          BNE     ASC             ;BRANCH IF BIT IS SET,
007204  104003                          ERROR                   ;RXCSR BIT6 FAILED TO SET,
007206  000410                          BR      ASD
007210  042777  000100  172166  ASC:    BIC     #BIT6,@RXCSR    ;CLEAR RXCSR BIT6
007216  032777  000100  172160          BIT     #BIT6,@RXCSR    ;SEE IF BIT IS CLEAR,
007224  001401                          BEQ     ASD
007226  104003                          ERROR                   ;RXCSR BIT6 FAILED TO CLEAR,
007230  052777  000100  172146  ASD:    BIS     #BIT6,@RXCSR    ;SET RXCSR BIT6,
007236  104011                          SRESET                  ;ISSUE RESET TO CLEAR BIT,
007240  104012                          SCOPE                   ;SCOPE
                                        TSTA    100,,AT,0
                                        TSTAA   AT,100,,\X+1+0,\X+2,\X+1
                                ;*******************************************************************
007242  000015                  AT15:   15                      ;TEST NUMBER                    *
007244  007270                          AT16                    ;ADDRESS OF NEXT TEST           *
007246  000144                          100.                    ;ITERATION COUNT                *
007250  007252                          ATA                     ;SCOPE ENTRY POINT              *
        000015                          X=X+1                   ;                               *
                                ;*******************************************************************
                                ;TEST THAT RXCSR BIT7 IS CLEAR AND CAN BE READ RELIABLY.
007252  032777  000200  172124  ATA:    BIT     #BIT7,@RXCSR    ;SEE IF RXCSR BIT7 IS CLEAR,
007260  001402                          BEQ     ATB             ;BRANCH IF BIT IS CLEAR,
007262  104003                          ERROR                   ;RXCSR BIT7 IS NOT CLEAR,
007264  104011                          SRESET                  ;RESET IF ERROR
007266  104012                  ATB:    SCOPE                   ;SCOPE
                                        TSTA    100,,AX,0
                                        TSTAA   AX,100,,\X+1+0,\X+2,\X+1
                                ;*******************************************************************
007270  000016                  AT16:   16                      ;TEST NUMBER                    *
007272  007316                          AT17                    ;ADDRESS OF NEXT TEST           *
007274  000144                          100.                    ;ITERATION COUNT                *
007276  007300                          AXA                     ;SCOPE ENTRY POINT              *
        000016                          X=X+1                   ;                               *
                                ;*******************************************************************
                                ;TEST THAT RXCSR BIT10 IS CLEAR AND CAN BE READ RELIABLY.
007300  032777  002000  172076  AXA:    BIT     #BIT10,@RXCSR   ;SEE IF RXCSR BIT10 IS CLEAR,
007306  001402                          BEQ     AXB             ;BRANCH IF BIT IS CLEAR,
007310  104003                          ERROR                   ;RXCSR BIT10 IS NOT CLEAR,
007312  104011                          SRESET                  ;RESET BIT IF ERROR
```

```
007314  104012                  AXB:    SCOPE                           ;SCOPE
                                        TSTA    100,,AY,CD
                                        TSTAA   AY,100',,\X+1+CD,\X+2,\X+1
                                ;*******************************************************************
007316  100017                  AT17:   100017                          ;TEST NUMBER                *
007320  007344                          AT20                            ;ADDRESS OF NEXT TEST       *
007322  000144                          100.                            ;ITERATION COUNT            *
007324  007326                          AYA                             ;SCOPE ENTRY POINT          *
        000017                          %8X+1                           ;                           *
                                ;*******************************************************************
                                ;TEST THAT RXCSR BIT11 IS CLEAR AND CAN BE READ RELIABLY;
007326  032777  004000  172050  AYA:    BIT     #BIT11,@RXCSR           ;SEE IF RXCSR BIT11 IS CLEAR;
007334  001402                  AYB:    BEQ     AYB                     ;BRANCH IF BIT IS CLEAR,
007336  104003                          ERROR                           ;RXCSR BIT11 IS NOT CLEAR;
007340  104011                          SRESET                          ;RESET BIT IF ERROR
007342  104012                  AYB:    SCOPE                           ;SCOPE
                                        TSTA    100,,AZ,CD
                                        TSTAA   AZ,100',,\X+1+CD,\X+2,\X+1
                                ;*******************************************************************
007344  100020                  AT20:   100020                          ;TEST NUMBER                *
007346  007372                          AT21                            ;ADDRESS OF NEXT TEST       *
007350  000144                          100.                            ;ITERATION COUNT            *
007352  007354                          AZA                             ;SCOPE ENTRY POINT          *
        000020                          %8X+1                           ;                           *
                                ;*******************************************************************
                                ;TEST THAT RXCSR BIT14 IS CLEAR AND CAN BE READ RELIABLY;
007354  032777  040000  172022  AZA:    BIT     #BIT14,@RXCSR           ;SEE IF RXCSR BIT14 IS CLEAR;
007362  001402                  AZB:    BEQ     AZB                     ;BRANCH IF BIT IS CLEAR,
007364  104003                          ERROR                           ;RXCSR BIT14 IS NOT CLEAR;
007366  104011                          SRESET                          ;RESET BIT IF ERROR
007370  104012                  AZB:    SCOPE                           ;SCOPE
                                        TSTA    100,,AAA,CD
                                        TSTAA   AAA,100',,\X+1+CD,\X+2,\X+1
                                ;*******************************************************************
007372  100021                  AT21:   100021                          ;TEST NUMBER                *
007374  007420                          AT22                            ;ADDRESS OF NEXT TEST       *
007376  000144                          100.                            ;ITERATION COUNT            *
007400  007402                          AAAA                            ;SCOPE ENTRY POINT          *
        000021                          %8X+1                           ;                           *
                                ;*******************************************************************
                                ;TEST THAT RXCSR BIT15 IS CLEAR AND CAN BE READ RELIABLY;
007402  032777  100000  171774  AAAA:   BIT     #BIT15,@RXCSR           ;SEE IF RXCSR BIT15 IS CLEAR;
007410  001402                  AAAB:   BEQ     AAAB                    ;BRANCH IF BIT IS CLEAR,
007412  104003                          ERROR                           ;RXCSR BIT15 IS NOT CLEAR;
007414  104011                          SRESET                          ;RESET BIT IF ERROR
007416  104012                  AAAB:   SCOPE                           ;SCOPE
                                ;
                                ;ALL PREVIOUS TESTS MUST HAVE BEEN RUN SUCCESSFULLY PRIOR
                                ;TO RUNNING THE FOLLOWING TESTS.  ALSO, THE JUMPER CONNECTOR
                                ;MUST BE INSERTED IN THE DL11-E CABLE IN PLACE OF THE MODEM.  COMMENTS
                                ;REFER TO OPERATION WITH JUMPER INSERTED.
                                ;
                                        TSTA    100,,AFB,0
                                        TSTAA   AFB,100',,\X+1+0,\X+2,\X+1
                                ;*******************************************************************
```

```
007420  000022                  AT22:   22                      ;TEST NUMBER                    *
007422  007474                          AT23                    ;ADDRESS OF NEXT TEST           *
007424  000144                          100.                    ;ITERATION COUNT                *
007426  007430                          AFBA                    ;SCOPE ENTRY POINT              *
        000022                          X=X+1                   ;                               *
                                ;*******************************************************************
                                ;TEST THAT CARRIER DETECT SETS AND CLEARS WHEN DATA TERMINAL
                                ;READY SETS AND CLEARS;
007430  052777  000002  171746  AFBA:   BIS     #BIT1,@RXCSR    ;SET DATA TERMINAL READY
007436  032777  010000  171740          BIT     #BIT12,@RXCSR   ;TEST CARRIER DETECT
007444  001002                          BNE     AFBB            ;SHOULD BE SET
007446  104003                          ERROR                   ;WASN'T
007450  000410                          BR      AFBC
007452  042777  000002  171724  AFBB:   BIC     #BIT1,@RXCSR    ;CLEAR DATA TERMINAL READY
007460  032777  010000  171716          BIT     #BIT12,@RXCSR   ;TEST CARRIER DETECT
007466  001401                          BEQ     AFBC
007470  104003                          ERROR                   ;WAS SET, ERROR
007472  104012                  AFBC:   SCOPE
                                        TSTA    100.,,AGB,0
                                        TSTAA   AGB,100.,,\X+1+0,\X+2,\X+1
                                ;*******************************************************************
007474  000023                  AT23:   23                      ;TEST NUMBER                    *
007476  007632                          AT24                    ;ADDRESS OF NEXT TEST           *
007500  000144                          100.                    ;ITERATION COUNT                *
007502  007504                          AGBA                    ;SCOPE ENTRY POINT              *
        000023                          X=X+1                   ;                               *
                                ;*******************************************************************
                                ;TEST THAT MODEM INTERRUPT (BIT 15) SETS WHEN CARRIER DETECT
                                ;CHANGES STATE, AND IS CLEARED WHEN RXCSR IS READ;
007504  042777  000002  171672  AGBA:   BIC     #BIT1,@RXCSR    ;CLEAR DATA TERMINAL READY
007512  017767  171666  172002          MOV     @RXCSR,RXCSRT   ;READ RXCSR
007520  032777  100000  171656          BIT     #BIT15,@RXCSR   ;TEST MODEM INTERRUPT
007526  001402                          BEQ     AGBB            ;WAS CLEAR  GO TO AGBB
007530  104003                          ERROR                   ;WASN'T CLEAR
007532  000436                          BR      AGBE            ;GO TO SCOPE
007534  052777  000002  171642  AGBB:   BIS     #BIT1,@RXCSR    ;SETTING DATA TERMINAL READY
                                                                ;CAUSES CARRIER DETECT TO SET
                                                                ;WHICH CAUSES MODEM INTERRUPT TO SET
007542  017767  171636  172032          MOV     @RXCSR,RXCSRT   ;MOVE RXCSR TO TEMPORARY LOCATION
007550  032767  100000  172024          BIT     #BIT15,RXCSRT   ;TEST MODEM INTERRUPT
007556  001002                          BNE     AGBC            ;SHOULD BE SET  GO TO AGBC
007560  104003                          ERROR                   ;WAS CLEAR
007562  000422                          BR      AGBE            ;GO TO SCOPE
007564  032777  100000  171612  AGBC:   BIT     #BIT15,@RXCSR   ;MODEM INTERRUPT BIT SHOULD
                                                                ;HAVE BEEN CLEARED
007572  001402                          BEQ     AGBD            ;IT WAS  GO TO AGBD
007574  104003                          ERROR                   ;IT WASN'T
007576  000414                          BR      AGBE            ;GO TO SCOPE
007600  042777  000002  171576  AGBD:   BIC     #BIT1,@RXCSR    ;CLEARING DATA TERMINAL READY
                                                                ;CAUSES CARRIER DETECT TO CLEAR
                                                                ;BUT MODEM INTERRUPT WILL SET
007606  017767  171572  171766          MOV     @RXCSR,RXCSRT   ;MOV RXCSR TO TEMPORARY LOCATION
007614  032767  100000  171760          BIT     #BIT15,RXCSRT   ;TEST MODEM INTERRUPT
007622  001002                          BNE     AGBE            ;SHOULD BE SET
007624  104003                          ERROR                   ;IT WASN'T
```

```
007626  000400                          BR      AGBE
007630  104012          AGBE1           SCOPE                           ;SCOPE
                                        TSTA    100,,AJB,0
                                        TSTAA   AJB,100,,\X+0,\X+2,\X+1
                        ;*******************************************************************
007632  000024          AT24            24                              ;TEST NUMBER
007634  007730                          AT25                            ;ADDRESS OF NEXT TEST
007636  000144                          100,                            ;ITERATION COUNT
007640  007642                          AJBA                            ;SCOPE ENTRY POINT
        000024                          X=X+1                           ;
                        ;*******************************************************************
                        ;TEST THAT CLEAR TO SEND (BIT13) SETS/CLEARS WHEN DATA TERMINAL
                        ;READY SETS/CLEARS;
007642  042777  000002  171534  AJBA1   BIC     #BIT1,@RXCSR            ;CLEAR DATA TERMINAL READY
007650  032777  020000  171526          BIT     #BIT13,@RXCSR           ;TEST CLEAR TO SEND
007656  001402                          BEQ     AJBB
007660  104003                          ERROR                           ;CLEAR TO SEND SHOULD BE CLEAR
007662  000421                          BR      AJBD
007664  052777  000002  171512  AJBB1   BIS     #BIT1,@RXCSR            ;SET DATA TERMINAL READY
007672  032777  020000  171504          BIT     #BIT13,@RXCSR           ;TEST CLEAR TO SEND
007700  001002                          BNE     AJBC                    ;BRANCH IF SET
007702  104003                          ERROR                           ;CLEAR TO SEND SHOULD BE SET
007704  000410                          BR      AJBD
007706  042777  000002  171470  AJBC1   BIC     #BIT1,@RXCSR            ;CLEAR DATA TERMINAL READY
007714  032777  020000  171462          BIT     #BIT13,@RXCSR           ;TEST CLEAR TO SEND
007722  001401                          BEQ     AJBD
007724  104003                          ERROR                           ;CLEAR TO SEND SHOULD BE CLEAR
007726  104012          AJBD1           SCOPE                           ;SCOPE
                                        TSTA    100,,AKB,0
                                        TSTAA   AKB,100,,\X+1+0,\X+2,\X+1
                        ;*******************************************************************
007730  000025          AT25            25                              ;TEST NUMBER
007732  010010                          AT26                            ;ADDRESS OF NEXT TEST
007734  000144                          100,                            ;ITERATION COUNT
007736  007740                          AKBA                            ;SCOPE ENTRY POINT
        000025                          X=X+1                           ;
                        ;*******************************************************************
                        ;TEST THAT RING (BIT 14 RXCSR) SETS WHEN REQUEST TO
                        ;SEND SETS AND CLEARS AND REPT CLEARS RING
007740  042777  000004  171436  AKBA1   BIC     #BIT2,@RXCSR            ;CLEAR REQUEST TO SEND
007746  052777  000004  171430          BIS     #BIT2,@RXCSR            ;SET REQUEST TO SEND
007754  032777  040000  171422          BIT     #BIT14,@RXCSR           ;TEST RING
007762  001001                          BNE     AKBC
007764  104003                          ERROR                           ;RING SHOULD BE SET
007766  042777  000004  171410  AKBC1   BIC     #BIT2,@RXCSR            ;CLEAR REQUEST TO SEND
007774  032777  040000  171402          BIT     #BIT14,@RXCSR           ;TEST RING
010002  001401                          BEQ     .+4                     ;SHOULD BE CLEAR
010004  104003                          ERROR
010006  104012                          SCOPE                           ;SCOPE
                                        TSTA    100,,AOB,0
                                        TSTAA   AOB,100,,\X+1+0,\X+2,\X+1
                        ;*******************************************************************
010010  000026          AT26            26                              ;TEST NUMBER
010012  010106                          AT27                            ;ADDRESS OF NEXT TEST
010014  000144                          100,                            ;ITERATION COUNT
```

```
010016  010020                    AOBA                    ;SCOPE ENTRY POINT
        000026                    X=X+1                   ;
                        ;*************************************************************
                        ;TEST THAT MODEM INTERRUPT (BIT 15 RXCSR) SETS WHEN RING SET,
010020  042777  000004  171356  AOBA1   BIC     #BIT2,@RXCSR    ;CLEAR REQUEST TO SEND
010026  032777  100000  171350          BIT     #BIT15,@RXCSR   ;TEST MODEM INTERRUPT BIT
010034  001402                        BEQ     AOBB
010036  104003                        ERROR
010040  000421                        BR      AOBD
010042  052777  000004  171334  AOBB1   BIS     #BIT2,@RXCSR    ;SET REQUEST TO SEND
010050  032777  100000  171326          BIT     #BIT15,@RXCSR   ;TEST MODEM INTERRUPT BIT
010056  001002                        BNE     AOBC
010060  104003                        ERROR
010062  000410                        BR      AOBD
010064  042777  000004  171312  AOBC1   BIC     #BIT2,@RXCSR    ;CLEAR REQUEST TO SEND
010072  032777  100000  171304          BIT     #BIT15,@RXCSR   ;TEST MODEM INTERRUPT BIT
010100  001401                        BEQ     AOBD
010102  104003                        ERROR
010104  104012                AOBD1   SCOPE                   ;SCOPE
                                      TSTA    100,,ALB,0
                                      TSTAA   ALB,100,,\X+1+2,\X+2,\X+1
                        ;*************************************************************
010106  000027                AT27!   27                      ;TEST NUMBER
010110  010204                        AT30                    ;ADDRESS OF NEXT TEST
010112  000144                        100.                    ;ITERATION COUNT
010114  010116                        ALBA                    ;SCOPE ENTRY POINT
        000027                        X=X+1                   ;
                        ;*************************************************************
                        ;TEST THAT SUPERVISORY RECEIVE DATA (BIT 10 RXCSR) SETS/CLEARS
                        ;WHEN SUPERVISORY XMIT DATA SETS/CLEARS.
010116  042777  000010  171260  ALBA1   BIC     #BIT3,@RXCSR    ;CLEAR SUPERVISOR XMIT DATA
010124  032777  002000  171252          BIT     #BIT10,@RXCSR   ;TEST SUPERVISORY RECEIVE DATA.
010132  001402                        BEQ     ALBB
010134  104003                        ERROR                   ;SHOULD HAVE BEEN CLEAR
010136  000421                        BR      ALBD
010140  052777  000010  171236  ALBB1   BIS     #BIT3,@RXCSR    ;SET SUPERVISORY XMIT DATA
010146  032777  002000  171230          BIT     #BIT10,@RXCSR   ;TEST SUPERVISORY RECEIVE DATA
010154  001002                        BNE     ALBC
010156  104003                        ERROR                   ;SHOULD HAVE BEEN SET
010160  000410                        BR      ALBD
010162  042777  000010  171214  ALBC1   BIC     #BIT3,@RXCSR    ;CLEAR SUPERVISORY XMIT DATA
010170  032777  002000  171206          BIT     #BIT10,@RXCSR   ;TEST SUPERVISORY RECEIVE DATA
010176  001401                        BEQ     ALBD
010200  104003                        ERROR                   ;SHOULD HAVE BEEN CLEAR
010202  104012                ALBD1   SCOPE                   ;SCOPE
                                      TSTA    100,,AMB,0
                                      TSTAA   AMB,100,,\X+1+2,\X+2,\X+1
                        ;*************************************************************
010204  000030                AT30!   30                      ;TEST NUMBER
010206  010324                        AT31                    ;ADDRESS OF NEXT TEST
010210  000144                        100.                    ;ITERATION COUNT
010212  010214                        AMBA                    ;SCOPE ENTRY POINT
        000030                        X=X+1                   ;
                        ;*************************************************************
                        ;TEST THAT SUP REC DATA TRANSISTIONS SET MODEM INTERRUPT
```

```
010214  142777  000010  171162  AMB4I   BIC     #BIT3,@RXCSR    ;CLEAR SUP REC
010222  052777  000010  171154          BIS     #BIT3,@RXCSR    ;SET SUP REC
010230  032777  100000  171146          BIT     #BIT15,@RXCSR   ;TEST MODEM INTERRUPT
010236  001002                          BNE     AMB8            ;MODEM INTERRUPT SHOULD BE SET
010240  104003                          ERROR
010242  000427                          BR      AMBE
010244  032777  100000  171132  AMB8I   BIT     #BIT15,@RXCSR   ;MODEM INTERRUPT SHOULD BE
010252  001402                          BEQ     AMBC            ;CLEARED BY PREVIOUS READ
010254  104003                          ERROR
010256  000421                          BR      AMBE
010260  042777  000010  171116  AMBCI   BIC     #BIT3,@RXCSR    ;LOAD TRANS OF SUP REC DATA
010266  032777  100000  171110          BIT     #BIT15,@RXCSR   ;TEST MODEM INTERRUPT
010274  001002                          BNE     AMBD            ;SHOULD BE SET
010276  104003                          ERROR
010300  000410                          BR      AMBE
010302  052777  000010  171074  AMBDI   BIS     #BIT3,@RXCSR    ;SET TRANS OF SUP REC DATA
010310  032777  100000  171066          BIT     #BIT15,@RXCSR   ;TEST MODEM INTERRUPT
010316  001001                          BNE     AMBE            ;SHOULD BE SET
010320  104003                          ERROR
010322  104012          AMBEI   SCOPE
                                TSTA    10.,ABA,CD
                                TSTAA   ABA,10.,\X+1+0,\X+2,\X+1
                        ;*****************************************************************
010324  100031          AT31:   100031                  ;TEST NUMBER                    *
010326  010430                  AT32                    ;ADDRESS OF NEXT TEST           *
010330  000012                  10.                     ;ITERATION COUNT                *
010332  010334                  ABAA                    ;SCOPE ENTRY POINT              *
        000031                  X=X+1                   ;                               *
                        ;*****************************************************************
                        ;TEST THAT RESET CLEARS ALL TXCSR BITS, AND SETS BIT 7 (READY)
010334  012767  000340  167434  ABAAI   MOV     #PRI7,PSW       ;SET PRIORITY 7,
010342  012777  177777  171040          MOV     #-1,@TXCSR      ;SET ALL POSSIBLE BITS IN TXCSR
010350  104011                          SRESET                  ;ISSUE RESET TO CLEAR BITS
010352  022777  000200  171030          CMP     #BIT7,@TXCSR    ;SEE IF ONLY BIT 7 IS SET,
010360  001422                          BEQ     ABAB            ;BRANCH IF ONLY BIT 7 IS SET
010362  017767  171022  171210          MOV     @TXCSR,TXCSRT   ;SAVE CONTENTS OF TXCSR
010370  012767  000200  171214          MOV     #BIT7,TEMP      ;MOVE EXPECTED TXCSR TO TEMP,
010376  004567  173442                  JSR     R5,OACNV        ;GO TO OCTAL TO ASCII CONVERT,
010402  001612                          TEMP                    ;SOURCE ADDR,
010404  014351                          ATXSB                   ;DESTINATION ADDR,
010406  000006                          6                       ;#OF DIGITS TO CONVERT,
010410  004567  173430                  JSR     R5,OACNV        ;GO TO OCTAL TO ASCII CONVERT,
010414  001600                          TXCSRT                  ;SOURCE ADDR,
010416  014366                          ATXWAS                  ;DESTINATION ADDR,
010420  000006                          6                       ;#OF DIGITS TO CONVERT,
010422  104015                          ERROR1                  ;RESET FAILED TO CLEAR ALL BITS EXCEPT
010424  014336                          ATXCSR                  ;BIT 7 - SEE PRINTOUT
010426  104012          ABABI   SCOPE                   ;SCOPE
                                TSTA    10.,ACA,0
                                TSTAA   ACA,10.,\X+1+0,\X+2,\X+1
                        ;*****************************************************************
012430  000032          AT32:   32                      ;TEST NUMBER                    *
012432  010670                  AT33                    ;ADDRESS OF NEXT TEST           *
012434  000012                  10.                     ;ITERATION COUNT                *
012436  010440                  ACAA                    ;SCOPE ENTRY POINT              *
```

```
                000032                          X=X+1                           ;
                                        ;*******************************************************************
                                        ;TEST THAT RESET CLEARS ALL RXCSR BITS EXCEPT DATA TERMINAL READY, RING
                                        ;CLEAR TO SEND,CARRIER DET
010440  012767  000340  167330  ACAAI   MOV     #PRTY7,@PSW             ;SET PRIORITY 7
010446  042777  000002  170730          BIC     #BIT1,@RXCSR            ;CLEAR DATA TERM,READY
010454  012777  177777  170722          MOV     #-1,@RXCSR              ;SET ALL POSSIBLE BITS IN RXCSR
010462  052777  000004  170720          BIS     #4,@TXCSR               ;SET MAINT BIT
010470  005077  170716                  CLR     @TXBUF                  ;TRANSMIT A CHAR
010474  104020                          TIMETX                          ;TIME OUT TX DONE
010476  104003                          ERROR                           ;ERROR DONE NOT SETTING
010500  012777  000001  170704          MOV     #1,@TXBUF               ;TRANSMIT ANOTHER CHAR'
010506  104017                          TIMERX                          ;TIME OUT RX DONE
010510  104003                          ERROR                           ;ERROR DONE NOT SETTING
010512  104011                          SRESET                          ;ISSUE RESET TO CLEAR BITS'
010514  017767  170664  171060          MOV     @RXCSR,RXCSRT           ;MOVE RXCSR CONTENTS TO RXCSRT
010522  022767  030002  171052          CMP     #30002,RXCSRT           ;SEE IF ONLY BITS 1,12,13 SET
010530  001417                          BEQ     ACAB                    ;BRANCH IF ONLY BITS 1,12,13 SET,
010532  012767  030002  171052          MOV     #30002,TEMP
010540  004567  173300                  JSR     %5,OACNV                ;GO TO OCTAL TO ASCII CONVERT,
010544  001612                          TEMP                            ;SOURCE ADDR,
010546  014410                          ARXSB                           ;DESTINATION ADDR',
010550  000006                          6                               ;#OF DIGITS TO CONVERT',
010552  004567  173266                  JSR     %5,OACNV                ;GO TO OCTAL TO ASCII CONVERT,
010556  001602                          RXCSRT                          ;SOURCE ADDR,
010560  014425                          ARXWAS                          ;DESTINATION ADDR,
010562  000006                          6                               ;#OF DIGITS TO CONVERT',
010564  104015                          ERROR1                          ;RESET FAILED TO CLEAR ALL BITS EXCEPT
010566  014375                          ARXCSR                          ;BIT 0', SEE ERROR PRINTOUT',
010570  042777  000002  170606  ACABI   BIC     #BIT1,@RXCSR            ;CLEAR DATA TERM, READY
010576  104012                          SCOPE                           ;SCOPE
                                        TSTA    10',,ADA,CD
                                        TSTAA   ADA,10',,\X+1+CD,\X+2,\X+1
                                        ;*******************************************************************
010600  100033                  AT331   100033                          ;TEST NUMBER                    *
010602  010662                          AT34                            ;ADDRESS OF NEXT TEST           *
010604  000012                          10',                            ;ITERATION COUNT                *
010606  010610                          ADAA                            ;SCOPE ENTRY POINT              *
        000033                          X=X+1                           ;                               *
                                        ;*******************************************************************
                                        ;TEST THAT LOADING TXBUF (TRANSMITTER BUFFER) CLEARS TXCSR BIT 7 (READY)
                                        ;AND WITHOUT MAINT SET THAT TXDONE SETS READY
010610  005077  170576          ADAAI   CLR     @TXBUF                  ;LOAD TXBUF
010614  104020                          TIMETX                          ;TIME OUT TX DONE
010616  104003                          ERROR                           ;ERROR, DONE NOT SETTING
010620  005077  170566                  CLR     @TXBUF                  ;LOAD TX BUF
010624  105777  170560                  TSTB    @TXCSR                  ;TEST TXCSR BIT 7 (READY BIT)
010630  100002                          BPL     ADAB                    ;BRANCH IF BIT NOT SET',
010632  104003                          ERROR                           ;ERROR, LOADING TXBUF FAILED TO CLEAR READY,
010634  000410                          BR      ADAC
010636  104020                  ADABI   TIMETX                          ;WAIT FOR DONE
010640  104003                          ERROR                           ;DONE NEVER SET
010642  000405                          BR      ADAC
010644  032777  000200  170536          BIT     #BIT7,@TXCSR
010652  001001                          BNE     .+4
```

```
010654  104003                          ERROR                   ;READY DID NOT SET
010656  104011                  ADAC1   SRESET
010660  104012                          SCOPE                   ;SCOPE;
                                        TSTA    1,;A1A,CD
                                        TSTAA   A1A,1',\X+1-CD,\X+2,\X-1
                                ;*******************************************************************************
010662  100034                  AT341   100034                  ;TEST NUMBER                                    *
010664  011204                          AT35                    ;ADDRESS OF NEXT TEST                           *
010666  000001                          1;                      ;ITERATION COUNT                                *
010670  010672                          A1AA                    ;SCOPE ENTRY POINT                              *
        000034                          X=X+1                   ;                                               *
                                ;*******************************************************************************
                                ;TEST THAT TRANSMIT SPEEDS ARE ARRANGED IN ASCENDING ORDER BY CHECKING THAT TIME
                                ;TO READY BIT (TXCSR BIT 7) DECREASES AS A HIGHER SPEED IS SELECTED;
010672  004767  174070          A1AA1   JSR     %7,DOTHIS       ;TEST IF THIS TEST SELECTED
010676  104001                          TYPES
010700  015654                          MSETTX
010702  015702                          MSETC
010704  016143                          MSØ
010706  177777                          -1
010710  000000                          HALT
010712  004767  000202                  JSR     %7,A1AS         ;OUTPUT CHAR AND TIME;
010716  016767  000260  170634          MOV     A1AST,CTRØ      ;MOVE ELAPSED TIME TO CTRØ;
010724  104000                          TYPE
010726  016153                          MS1
010730  000000                          HALT
010732  004767  000162                  JSR     %7,A1AS         ;OUTPUT CHAR AND TIME;
010736  016767  000240  170616          MOV     A1AST,CTR1      ;MOVE ELAPSED TIME TO CTR1;
010744  104000                          TYPE
010746  016163                          MS2
010750  000000                          HALT
010752  004767  000142                  JSR     %7,A1AS         ;OUTPUT CHAR AND TIME;
010756  016767  000220  170600          MOV     A1AST,CTR2      ;MOVE ELAPSED TIME TO CTR2;
010764  104000                          TYPE
010766  016173                          MS3
010770  000000                          HALT
010772  004767  000122                  JSR     %7,A1AS         ;OUTPUT CHAR AND TIME;
010776  016767  000200  170562          MOV     A1AST,CTR3      ;MOVE ELAPSED TIME TO CTR3;
011004  104000                          TYPE
011006  016203                          MS4
011010  000000                          HALT
011012  004767  000102                  JSR     %7,A1AS         ;OUTPUT CHAR AND TIME
011016  016767  000160  170544          MOV     A1AST,CTR4      ;MOVE ELAPSED TIME TO CTR4
011024  104000                          TYPE
011026  016213                          MS5
011030  000000                          HALT
011032  004767  000062                  JSR     %7,A1AS         ;OUTPUT CHAR AND TIME
011036  016767  000140  170526          MOV     A1AST,CTR5      ;MOVE ELAPSED TIME TO CTR5
011044  104000                          TYPE
011046  016223                          MS6
011050  000000                          HALT
011052  004767  000042                  JSR     %7,A1AS         ;OUTPUT CHAR AND TIME
011056  016767  000120  170510          MOV     A1AST,CTR6      ;MOVE ELAPSED TIME TO CTR6
011064  104000                          TYPE
011066  016233                          MS7
```

```
011070  000000                        HALT
011072  004767  000022                JSR     %7,AIAS         ;OUTPUT CHAR AND TIME
011076  016767  000100  170472        MOV     AIAST,CTR7      ;MOVE ELAPSED TIME TO CTR7
011104  004767  002276                JSR     %7,CMPT         ;CHECK THAT CTR0 THROUGH CTR7 CONTAIN
011110  000402                        BR      AIAF            ;DESCENDING VALUES
011112  104015                        ERROR1                  ;TRANSMIT SPEEDS NOT ARRANGED IN
011114  014435                        ETXTIM                  ;ASCENDING ORDER,
011116  104012              AIAF1:    SCOPE                   ;SCOPE
                            ;
011120  005067  000036      AIAS1:    CLR     AIAST           ;CLEAR ELAPSED TIME COUNTER,
011124  105777  170260                TSTB    @TXCSR          ;WAIT FOR TX READY,
011130  100375                        BPL     .-4
011132  005077  170254                CLR     @TXBUF
011136  105777  170246                TSTB    @TXCSR
011142  100375                        BPL     .-4
011144  005077  170242                CLR     @TXBUF          ;LOAD TXBUF,
011150  004767  000014      AIASA1:   JSR     %7,TIME         ;WAIT 75 US
011154  005267  000022                INC     AIAST           ;INCREMENT ELAPSED TIME COUNTER,
011160  105777  170224                TSTB    @TXCSR          ;READY SET?
011164  100371                        BPL     AIASA           ;BRANCH IF READY NOT SET,
011166  000207                        RTS     %7              ;EXIT,
                            ;
011170  012700  000017      TIME1:    MOV     #15.,%0
011174  005300              TIM11:    DEC     %0
011176  001376                        BNE     TIM11
011200  000207                        RTS     %7
0112^2  000000              AIAST:    OPEN
                                      TSTA    10.,ALA,0
                                      TSTAA   ALA,10.,\X+1+0,\X+2,\X+1
                            ;*****************************************************************************
011204  000035              AT351:    35                      ;TEST NUMBER                               *
011206  011260                        AT36                    ;ADDRESS OF NEXT TEST                      *
011210  000012                        10.                     ;ITERATION COUNT                           *
011212  011214                        ALAA                    ;SCOPE ENTRY POINT                         *
        000035                        X=X+1                   ;                                          *
                            ;*****************************************************************************
                            ;TEST THAT OUTPUTTING A CHARACTER WITH THE MAINTENANCE BIT SET (TXCSR BIT 2)
                            ;RESULTS IN DONE BIT SETTING (RXCSR BIT 7) NO LATER THAN 500 MSECS, AND
                            ;THAT RESET INSTRUCTION CLEARS THE DONE BIT
011214  052777  000004  170166  ALAA1:  BIS     #BIT2,@TXCSR    ;SET MAINTENANCE (TXCSR BIT 2)
011222  005077  170164              CLR     @TXBUF          ;LOAD TXBUF
011226  104016                      DELAY                   ;WAIT 500 MSECS,
011230  000764                      500.
011232  105777  170146              TSTB    @RXCSR          ;SEE IF DONE BIT IS SET
011236  100402                      BMI     ALAB            ;BRANCH IF DONE BIT IS SET
011240  104003                      ERROR                   ;DONE BIT FAILED TO SET
011242  000405                      BR      ALAC
011244  104011              ALAB1:  SRESET                  ;ISSUE RESET TO CLEAR DONE BIT
011246  105777  170132              TSTB    @RXCSR          ;SEE IF DONE BIT IS CLEARED
011252  100001                      BPL     ALAC            ;BRANCH IF DONE BIT IS CLEARED
011254  104003                      ERROR                   ;RESET FAILED TO CLEAR DONE BIT
011256  104012              ALAC1:  SCOPE                   ;SCOPE
                                    TSTA    100.,AMA,CD
                                    TSTAA   AM[illegible]00.,,\X+1+CD,\X+2,\X+1
                            ;*****************************************************************************
```

```
011240  100036                  AT361   100036                  ;TEST NUMBER                    *
011242  011324                          AT37                    ;ADDRESS OF NEXT TEST           *
011244  000144                          100.                    ;ITERATION COUNT                *
011246  011270                          AMAA                    ;SCOPE ENTRY POINT              *
        000036                          X=X+1                   ;                               *
                        ;*****************************************************************************
                        ;TEST THAT DONE BIT (RXCSR BIT 7) IS CLEARED BY READING RXBUF,
                        ;DONE SET BY OUTPUTTING CHARACTER WITH MAINTENANCE BIT SET (TXCSR BIT 2)
011270  052777  000004  170112  AMAA1   BIS     #BIT2,@TXCSR    ;SET MAINTENANCE BIT (TXCSR BIT 2)
011276  005077  170110          CLR     @TXBUF                  ;LOAD TXBUF
011302  104017                  TIMERX                          ;WAIT FOR DONE BIT TO SET;
011304  104003                  ERROR
011306  005777  170074          TST     @RXBUF                  ;READ RXBUF TO CLEAR DONE BIT
011312  105777  170066          TSTB    @RXCSR                  ;SEE IF DONE BIT IS CLEAR
011316  100001                  BPL     AMAC                    ;BRANCH IF DONE BIT IS CLEAR
011320  104003                  ERROR                           ;READING RXBUF FAILED TO CLEAR DONE BIT
011322  104012          AMAC1   SCOPE                           ;SCOPE
                                TSTA    100.,,AQA,CD
                                TSTAA   AQA;100.,,\X+1*CD,\X+2,\X+1
                        ;*****************************************************************************
011324  100037          AT371   100037                  ;TEST NUMBER                            *
011326  011434                  AT40                    ;ADDRESS OF NEXT TEST                   *
011330  000144                  100.                    ;ITERATION COUNT                        *
011332  011340                  AQAA                    ;SCOPE ENTRY POINT                      *
        000037                  X=X+1                   ;                                       *
                        ;*****************************************************************************
                        ;TEST THAT RECEIVER ACTIVE SETS WHEN CHAR STARTS AND
                        ;CLEARS WHEN RECEIVER DONE SETS
011334  004767  171070          JSR     %7,CDINIT               ;INIT IF C=D DEVICE
011340  052777  000004  170042  AQAA1   BIS     #BIT2,@TXCSR    ;SET MAINT
011346  005077  170040          CLR     @TXBUF                  ;TRANSMIT CHAR
011352  005067  170234          CLR     TEMP                    ;CLEAR BUSY INDICATOR
011356  032777  004000  170020  AQAB1   BIT     #BIT11,@RXCSR   ;IS RECEIVER ACTIVE SET
011364  001402                  BEQ     AQAB1                   ;BRANCH IF CLEAR
011366  005267  170220          INC     TEMP                    ;YES, REMEMBER THAT
011372  105777  170006          AQAB11  TSTB    @RXCSR          ;SEE IF DONE SET
011376  100367                  BPL     AQAB
011400  026727  170206  000000  CMP     TEMP,#0                 ;DID RECEIVER ACTIVE SET
011406  001002                  BNE     AQAC
011410  104003                  ERROR                           ;RECEIVER ACTIVE NEVER SET
011412  000405                  BR      AQAD
011414  032777  004000  167762  AQAC1   BIT     #BIT11,@RXCSR   ;DID DONE CLEAR ACTIVE
011422  001401                  BEQ     AQAD
011424  104003                  ERROR                           ;NO, RECEIVER ACTIVE DID NOT CLEAR
011426  005777  167754          AQAD1   TST     @RXBUF          ;CLEAR RX DONE
011432  104012                  SCOPE
                                TSTA    1,,AQA,0
                                TSTAA   AQA,1,,\X+1+0,\X+2,\X+1
                        ;*****************************************************************************
011434  000040          AT401   40                      ;TEST NUMBER                            *
011436  011736                  AT41                    ;ADDRESS OF NEXT TEST                   *
011440  000001                  1.                      ;ITERATION COUNT                        *
011442  011444                  AQAA                    ;SCOPE ENTRY POINT                      *
        000040                  X=X+1                   ;                                       *
                        ;*****************************************************************************
```

```
                                        ;TEST THAT RECEIVE SPEEDS ARE ARRANGED IN ASCENDING ORDER BY LOOKING THAT TIME
                                        ;ELAPSED TO DONE BIT SETTING (RXCSR BIT 7) DECREASES AS A HIGHER SPEED
                                        ;THIS IS NOT DONE IN MAINTENANCE MODE TX AND RX
                                        ;PDTS MUST BE STEPPED TOGETHER
                                        ;IS SELECTED,
011444  004767  174116          AQAA1:  JSR     %7,DOTHIS       ;CHECK IF THIS TEST TO BE DONE
011450  104001                          TYPES
011452  015626                          MSETRX
011454  015702                          MSETC
011456  016143                          MS0
011460  177777                          -1
011462  000000                          HALT
011464  004767  000202                  JSR     %7,AQAS         ;OUTPUT CHARACTER AND TIME DONE BIT
011470  016767  000240  170062          MOV     AQAST,CTR0      ;MOVE ELAPSED TIME TO CTR0
011476  104000                          TYPE
011500  016153                          MS1
011502  000000                          HALT
011504  004767  000162                  JSR     %7,AQAS         ;OUTPUT CHARACTER AND TIME DONE BIT
011510  016767  000220  170044          MOV     AQAST,CTR1      ;MOVE ELAPSED TIME TO CTR1
011516  104000                          TYPE
011520  016163                          MS2
011522  000000                          HALT
011524  004767  000142                  JSR     %7,AQAS         ;OUTPUT CHARACTER AND TIME DONE BIT,
011530  016767  000200  170026          MOV     AQAST,CTR2      ;MOVE ELAPSED TIME TO CTR2',
011536  104000                          TYPE
011540  016173                          MS3
011542  000000                          HALT
011544  004767  000122                  JSR     %7,AQAS         ;OUTPUT CHARACTER AND TIME DONE BIT
011550  016767  000160  170010          MOV     AQAST,CTR3      ;MOVE ELAPSED TIME TO CTR3',
011556  104000                          TYPE
011560  016203                          MS4
011562  000000                          HALT
011564  004767  000102                  JSR     %7,AQAS
011570  016767  000140  167772          MOV     AQAST,CTR4
011576  104000                          TYPE
011600  016213                          MS5
011602  000000                          HALT
011604  004767  000062                  JSR     %7,AQAS
011610  016767  000120  167754          MOV     AQAST,CTR5
011616  104000                          TYPE
011620  016223                          MS6
011622  000000                          HALT
011624  004767  000042                  JSR     %7,AQAS
011630  016767  000100  167736          MOV     AQAST,CTR6
011636  104000                          TYPE
011640  016233                          MS7
011642  000000                          HALT
011644  004767  000022                  JSR     %7,AQAS
011650  016767  000060  167720          MOV     AQAST,CTR7
011656  004767  001524                  JSR     %7,CMPT         ;CHECK THAT CTR0 THROUGH CTR3 CONTAIN
011662  000402                          BR      AQAB            ;DESCENDING VALUES',
011664  104015                          ERROR1                  ;RECEIVE SPEEDS NOT ARRANGED IN
011666  014477                          ERXTIM                  ;ASCENDING ORDER,
011670  104012                  AQAB1:  SCOPE                   ;SCOPE
                                        ;
```

```
011672  005067  000036          AQAS1   CLR     AQAST           ;CLEAR ELAPSED TIME COUNTER AQAST
011676  105777  167506                  TSTB    @TXCSR          ;WAIT FOR TX READY;
011702  100375                          BPL     .-4
011704  005777  167476                  TST     @RXBUF          ;CLEAR DONE BIT IF SET
011710  005077  167476                  CLR     @TXBUF          ;LOAD TXBUF
011714  004767  177250          AQASA1  JSR     %7,TIME
011720  005267  000010                  INC     AQAST           ;INCREMENT ELAPSED TIME COUNTER
011724  105777  167454                  TSTB    @RXCSR          ;DONE SET?
011730  100371                          BPL     AQASA           ;BRANCH IF DONE NOT SET
011732  000207                          RTS     %7              ;EXIT
011734  000000          AQAST:  OPEN                            ;ELAPSED TIME COUNTER
                                TSTA    10',,ARA,CD
                                TSTAA   ARA,10',,\X+1+CD,\X+2,\X+1
                        ;*****************************************************************
011736  100041          AT411   100041                          ;TEST NUMBER                *
011740  012102                  AT42                            ;ADDRESS OF NEXT TEST       *
011742  000012                  10'                             ;ITERATION COUNT            *
011744  011746                  ARAA                            ;SCOPE ENTRY POINT          *
        000041                  X=X+1                           ;                           *
                        ;*****************************************************************
                        ;TEST CORRECT OPERATION OF DATA OVERRUN BIT (RXBUF BIT 14)
011746  004767  000110  ARAA1   JSR     %7,ARAS         ;OUTPUT CHARACTER AND WAIT 500 MSECS
011752  004767  000104          JSR     %7,ARAS         ;OUTPUT CHARACTER AND WAIT 500 MSECS
011756  017767  167424  167620  MOV     @RXBUF,RXBUFT   ;SAVE RXBUF CONTENTS & CLEAR DONE
011764  032767  040000  167612  BIT     #BIT14,RXBUFT   ;SEE IF DATA OVERRUN BIT WAS SET
011772  001002                  BNE     .+6             ;BRANCH IF BIT WAS SET
011774  104003                  ERROR
011776  104012                  SCOPE
012000  005767  167600          TST     RXBUFT          ;SEE THAT ERROR BIT WAS SET (RXBUF BIT 15)
012004  100402                  BMI     .+6
012006  104003                  ERROR                   ;ERROR BIT FAILED TO SET WHEN OVERRUN SET
012010  104012                  SCOPE
012012  032777  040000  167566  BIT     #BIT14,@RXBUF   ;SEE THAT DATA OVERRUN WAS NOT
                                                        ;CLEARED WHEN RXBUF WAS READ
012020  001002                  BNE     .+6             ;BRANCH IF SET
012022  104003                  ERROR                   ;READING RXBUF CLEARED DATA OVERRUN
012024  104012                  SCOPE
012026  004767  000030          JSR     %7,ARAS
012032  032777  100000  167546  BIT     #BIT15,@RXBUF   ;OUTPUT CHAR +WAIT 500MS
012040  001402                  BEQ     .+6             ;TEST THAT ERROR CLEARED
012042  104003                  ERROR
012044  104012                  SCOPE
012046  032777  040000  167532  BIT     #BIT14,@RXBUF   ;TEST THAT OVERRUN CLEARED
012054  001401                  BEQ     .+4
012056  104003                  ERROR
012060  104012                  SCOPE                   ;SCOPE
012062  052777  000004  167520  ARAS1   BIS     #BIT2,@TXCSR    ;SET MAINTENANCE BIT
012070  005077  167516          CLR     @TXBUF          ;LOAD TXBUF
012074  104016                  DELAY                   ;DELAY 500 MSECS
012076  000764                  500.
012100  000207                  RTS     %7              ;EXIT
                                TSTA    10',,ATA,CD
                                TSTAA   ATA,10',,\X+1+CD,\X+2,\X+1
                        ;*****************************************************************
012102  100042          AT421   100042                          ;TEST NUMBER                *
```

```
012174  012170                          AT43                    ;ADDRESS OF NEXT TEST                  *
012176  000012                          10.                     ;ITERATION COUNT                       *
012110  012122                          ATAA                    ;SCOPE ENTRY POINT                     *
        000042                          X=X+1                   ;                                      *
                                ;*****************************************************************
                                ;TEST THAT TRANSMITTER IS ABLE TO INTERRUPT,  IF THE INTERRUPT IS SERVICED,
                                ;IT WILL HAVE OCCURRED AT THE CORRECT VECTOR,
012112  004767  171122                  JSR     7,OVRLAY        ;GO TO OVER LAY ROUTINE
012116  104007                          STTXV                   ;SET TX INTERRUPT SERVICE
012120  012156                          ATAC                    ;TO ATAC
012122  042777  000100  167260  ATAA:   BIC     #BIT6,@TXCSR    ;DISABLE TX INTERRUPT
012130  005067  165942                  CLR     PSW             ;SET PROCESSOR PRIORITY TO 0
012134  052777  000100  167246          BIS     #BIT6,@TXCSR    ;ENABLE TX INTERRUPT
012142  000240                          NOP
012144  104003                          ERROR                   ;READY DID NOT CAUSE AN INTERRUPT
012146  042777  000100  167234          BIC     #BIT6,@TXCSR
012154  104012                  ATAB:   SCOPE                   ;SCOPE
012156  042777  000100  167224  ATAC:   BIC     #BIT6,@TXCSR    ;HERE IF INT, DISABLE TX INT
012164  022626                          POPSP2
012166  000772                          BR      ATAB
                                        TSTA    1000,,AUA,CD
                                        TSTAA   AUA,1000,,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************
012170  100043                  AT43:   100043                  ;TEST NUMBER                           *
012172  012246                          AT44                    ;ADDRESS OF NEXT TEST                  *
012174  001750                          1000.                   ;ITERATION COUNT                       *
012176  012204                          AUAA                    ;SCOPE ENTRY POINT                     *
        000043                          X=X+1                   ;                                      *
                                ;*****************************************************************
                                ;TEST THAT READY DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR IS
                                ;AT THE SAME PRIORITY AS THE TRANSMITTER INTERRUPT REQUEST LEVEL
012200  104007                          STTXV                   ;SET TX INTERRUPT SERVICE TO
012202  012240                          AUAC                    ;
012204  016767  167212  165564  AUAA:   MOV     TXLVL,PSW       ;SET PROCESSOR PRIORITY SAME AS TX PRIORITY
012212  042777  000100  167170          BIC     #BIT6,@TXCSR
012220  052777  000100  167162          BIS     #BIT6,@TXCSR    ;ENABLE TX INTERRUPTS
012226  000240                          NOP
012230  042777  000100  167152  AUAB:   BIC     #BIT6,@TXCSR    ;OK IF NO INTERRUPT OCCURS,  DISABLE INTERRUPTS
012236  104012                          SCOPE                   ;SCOPE
012240  022626                  AUAC:   POPSP2                  ;HERE IF INTERRUPT OCCURS, POP STOCK TWICE
012242  104003                          ERROR                   ;TX INTERRUPTED WITH PROCESSOR AT SAME
012244  000771                          BR      AUAB            ;PRIORITY AS THE TRANSMITTER
                                        TSTA    10.,,AVA,CD
                                        TSTAA   AVA,10.,,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************
012246  100044                  AT44:   100044                  ;TEST NUMBER                           *
012250  012332                          AT45                    ;ADDRESS OF NEXT TEST                  *
012252  000012                          10.                     ;ITERATION COUNT                       *
012254  012262                          AVAA                    ;SCOPE ENTRY POINT                     *
        000044                          X=X+1                   ;                                      *
                                ;*****************************************************************
                                ;TEST THAT TRANSMITTER INTERRUPTS WHEN PROCESSOR IS AT PRIORITY ONE LEVEL
                                ;LOWER THAN THE TRANSMITTER INTERRUPT PRIORITY,
012256  104007                          STTXV                   ;SET TX INTERRUPT SERVICE TO AVAB
012260  012320                          AVAB
```

```
012262  042777  000100  167120  AVAAI:  BIC     #BIT6,@TXCSR    ;DISABLE TX INTERRUPTS
012270  016767  167126  165500          MOV     TXLVL,PSW       ;SET PROCESSOR PRIORITY TO ONE LEVEL
012276  162767  000040  165472          SUB     #40,PSW         ;LOWER THAN TX PRIORITY
012304  052777  000100  167076          BIS     #BIT6,@TXCSR    ;ENABLE TX INTERRUPTS
012312  000240                          NOP
012314  104003                          ERROR                   ;TX FAILED TO INTERRUPT
012316  000401                          BR      AVAC
012320  022626                  AVABI:  POPSP2                  ;HERE IF INTERRUPT OCCURS, POP STACK TWICE
012322  042777  000100  167060  AVACI:  BIC     #BIT6,@TXCSR    ;DISABLE TX INTERRUPTS
012330  104012                          SCOPE                   ;SCOPE
                                        TSTA    100,,AWA,CD
                                        TSTAA   AWA,100,,\X+1+CD,\X+2,\X+1
                        ;*****************************************************************************
012332  100045                  AT45:   100045                  ;TEST NUMBER                        *
012334  012430                          AT46                    ;ADDRESS OF NEXT TEST               *
012336  000144                          100.                    ;ITERATION COUNT                    *
012340  012342                          AWAA                    ;SCOPE ENTRY POINT                  *
        000045                          X=X+1                   ;                                   *
                        ;*****************************************************************************
                        ;TEST THAT TRANSMITTER DOES NOT REINTERRUPT AFTER THE INITIAL INTERRUPT HAS
                        ;OCCURRED AND HAS BEEN SERVICED.
012342  104007                  AWAAI:  STTXV                   ;SET TX INTERRUPT SERVICE TO AWAC
012344  012402                          AWAC
012346  042777  000100  167034          BIC     #BIT6,@TXCSR    ;DISABLE TX INTERRUPTS
012354  005067  165416                  CLR     PSW             ;SET PROCESSOR PRIORITY TO 0
012360  052777  000100  167022          BIS     #BIT6,@TXCSR    ;ENABLE TX INTERRUPTS
012366  000240                          NOP
012370  104003                          ERROR                   ;TRANSMITTER FAILED TO INTERRUPT
012372  042777  000100  167010  AWABI:  BIC     #BIT6,@TXCSR    ;DISABLE TX INTERRUPTS
012400  104012                          SCOPE                   ;SCOPE
012402  012777  012422  167010  AWACI:  MOV     #AWAE,@TXVTR    ;HERE IF INTERRUPT OCCURS, CHANGE EXIT
012410  012716  012416                  MOV     #AWAD,@X6       ;POINTER TO AWAD AND EXIT INTERRUPT
012414  000002                          RTI
012416  000240                  AWADI:  NOP                     ;OK IF NO INTERRUPT REOCCURS.
012420  000764                          BR      AWAB
012422  022626                  AWAEI:  POPSP2                  ;HERE IF INTERRUPT REOCCURS
012424  104003                          ERROR                   ;TX REINTERRUPTED AFTER RTI
012426  000761                          BR      AWAB
                                        TSTA    10.,AXA,CD
                                        TSTAA   AXA,10.,\X+1+CD,\X+2,\X+1
                        ;*****************************************************************************
012430  100046                  AT46:   100046                  ;TEST NUMBER                        *
012432  012514                          AT47                    ;ADDRESS OF NEXT TEST               *
012434  000012                          10.                     ;ITERATION COUNT                    *
012436  012454                          AXAA                    ;SCOPE ENTRY POINT                  *
        000046                          X=X+1                   ;                                   *
                        ;*****************************************************************************
                        ;TEST THAT RECEIVER DONE BIT IS ABLE TO INTERRUPT, IF THE INTERRUPT IS
                        ;SERVICED IT WILL HAVE OCCURRED AT THE CORRECT VECTOR.
012440  004767  170574          JSR     7,OVRLAY                ;GO TO OVERLAY ROUTINE
012444  104006                  STRXV                           ;SET RX INTERRUPT SERIVCE TO AXAB
012446  012502                  AXAB
012450  004767  000712          JSR     %7,$TRXD                ;SET RX DONE BIT
012454  042777  000100  166722  AXAAI:  BIC     #BIT6,@RXCSR    ;DISABLE RX INTERRUPTS
012462  005067  165310          CLR     PSW                     ;SET PROCESSOR PRIORITY TO 0
```

```
012466  052777  000100  166710          BIS     #BIT6,@RXCSR    ;ENABLE RX INTERRUPTS
012474  000240                          NOP
012476  104003                          ERROR                   ;RX FAILED TO INTERRUPT
012500  000401                          BR      AXAC
012502  022626                  AXAB:   POPSP2                  ;HERE IF INTERRUPT OCCURS
012504  042777  000100  166672  AXAC:   BIC     #BIT6,@RXCSR    ;DISABLE INT EN
012512  104012                          SCOPE                   ;SCOPE
                                        TSTA    10.,,AX1,0
                                        TSTAA   AX1,10.,,\X+1+0,\X+2,\X+1
                        ;*****************************************************************
012514  000047                  AT47:   47                      ;TEST NUMBER                *
012516  012576                          AT50                    ;ADDRESS OF NEXT TEST       *
012520  000012                          10.                     ;ITERATION COUNT            *
012522  012534                          AX1A                    ;SCOPE ENTRY POINT          *
        000047                          X=X+1                   ;                           *
                        ;*****************************************************************
                        ;TEST THAT MODEM INTERRUPT BIT IS ABLE TO INTERRUPT, IF THE INTERRUPT IS
                        ;SERVICED IT WILL HAVE OCCURRED AT THE CORRECT VECTOR,
012524  004767  170510                  JSR     7,OVRLAY        ;GO TO OVERLAY ROUTINE
012530  104006                          STRXV                   ;SET RX INTERRUPT SERIVCE TO AXAB
012532  012564                          AX1B
012534  042777  000044  166642  AX1A:   BIC     #44,@RXCSR      ;DISABLE MODEM INTERRUPTS
012542  005067  165250                  CLR     PSW             ;SET PROCESSOR PRIORITY TO 0
012546  052777  000044  166630          BIS     #44,@RXCSR      ;ENABLE MODEM INTERRUPTS,RQ TO SND
012554  104016                          DELAY
012556  000005                          5.
012560  104003                          ERROR                   ;MODEM FAILED TO INTERRUPT
012562  000401                          BR      AX1C
012564  022626                  AX1B:   POPSP2                  ;HERE IF INTERRUPT OCCURS
012566  042777  000040  166610  AX1C:   BIC     #BIT5,@RXCSR    ;DISABLE INT EN
012574  104012                          SCOPE
                                        TSTA    1000.,,AYA,CD
                                        TSTAA   AYA,1000.,,\X+1+CD,\X+2,\X+1
                        ;*****************************************************************
012576  100050                  AT50:   100050                  ;TEST NUMBER                *
012600  012600                          AT51                    ;ADDRESS OF NEXT TEST       *
012602  001750                          1000.                   ;ITERATION COUNT            *
012604  012616                          AYAA                    ;SCOPE ENTRY POINT          *
        000050                          X=X+1                   ;                           *
                        ;*****************************************************************
                        ;TEST THAT RECEIVER DONE BIT DOES NOT CAUSE AN INTERRUPT WHEN THE PROCESSOR
                        ;IS AT THE SAME PRIORITY LEVEL AS THE RECEIVER INTERRUPT REQUEST LEVEL
012606  104006                          STRXV                   ;SET RX INTERRUPT SERVICE TO AYAC
012610  012632                          AYAC
012612  004767  000350                  JSR     %7,STRXD        ;SET RX DONE BIT
012616  042777  000100  166560  AYAA:   BIC     #BIT6,@RXCSR    ;DISABLE RX INTERRUPTS
012624  016767  166566  165144          MOV     RXLVL,PSW       ;SET PROCESSOR PRIORITY SAME AS RECEIVER'S
012632  052777  000100  166544          BIS     #BIT6,@RXCSR    ;ENABLE RX INTERRUPTS
012640  000240                          NOP
012642  042777  000100  166534  AYAB:   BIC     #BIT6,@RXCSR    ;OK IF NO INTERRUPT, DISABLE RX INTERRUPTS
012650  104012                          SCOPE                   ;SCOPE
012652  022626                  AYAC:   POPSP2                  ;HERE IF INTERRUPT OCCURS, POP STOCK TWICE
012654  104003                          ERROR                   ;RX INTERRUPTED WITH PROCESOR AT SAME
012656  000771                          BR      AY              ;PRIORITY AS THE RECEIVER
                                        TSTA    10.,,AYA,CD
```

```
                                        TSTAA   AZA,10',,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************************
012660  100051                  AT511   100051                  ;TEST NUMBER                                *
012662  012750                          AT52                    ;ADDRESS OF NEXT TEST                       *
012664  000012                          10',                    ;ITERATION COUNT                            *
012666  012700                          AZAA                    ;SCOPE ENTRY POINT                          *
        000051                          X=X+1                   ;                                           *
                                ;*****************************************************************************
                                ;TEST THAT RECEIVER DONE BIT CAUSES INTERRUPT WHEN PROCESSOR IS AT PRIORITY
                                ;ONE LEVEL LOWER THAN THE RECEIVER'S INTERRUPT REQUEST LEVEL
012670  104006                          STRXV                   ;SET RX INTERRUPT TO AZAB
012672  012736                          AZAB
012674  004767  000466                  JSR     %7,STRXD        ;SET RX DONE BIT
012700  042777  000100  166476  AZAA1   BIC     #BIT6,@RXCSR    ;DISABLE RX INTERRUPTS
012706  016767  166504  165062          MOV     RXLVL,PSW       ;SET PROCESSOR PRIORITY ONE LEVEL
012714  162767  000040  165054          SUB     #40,@PSW        ;LOWER THAN RECEIVER'S PRIORITY
012722  052777  000100  166454          BIS     #BIT6,@RXCSR    ;ENABLE RX INTERRUPTS
012730  000240                          NOP
012732  104003                          ERROR                   ;RX FAILED TO INTERRUPT WITH PROCESSOR AT
012734  000401                          BR      AZAC            ;PRIORITY ONE LEVEL LOWER THAN RECEIVER'S
012736  022626                  AZAB1   POPSP2                  ;HERE IF INTERRUPT OCCURS
012740  042777  000100  166436  AZAC1   BIC     #BIT6,@RXCSR    ;DISABLE RX INTERRUPTS
012746  104012                          SCOPE                   ;SCOPE
                                        TSTA    100,,AAB,CD
                                        TSTAA   AAB,100,,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************************
012750  100052                  AT521   100052                  ;TEST NUMBER                                *
012752  013046                          AT53                    ;ADDRESS OF NEXT TEST                       *
012754  000144                          100,                    ;ITERATION COUNT                            *
012756  012764                          AABA                    ;SCOPE ENTRY POINT                          *
        000052                          X=X+1                   ;                                           *
                                ;*****************************************************************************
                                ;TEST THAT RECEIVER DOES NOT INTERRUPT AFTER THE INITIAL INTERRUPT HAS
                                ;OCCURED AND DONE BIT HAS NOT BEEN CLEARED
012760  004767  000402                  JSR     %7,STRXD        ;SET RX DONE BIT
012764  104006                  AABA1   STRXV                   ;SET RX INTERRUPT SERVICE TO AABC
012766  013020                          AABC
012770  042777  000100  166406          BIC     #BIT6,@RXCSR    ;DISABLE RX INTERRUPTS
012776  052777  000100  166400          BIS     #BIT6,@RXCSR    ;ENABLE RX INTERRUPTS
013004  000240                          NOP
013006  104003                          ERROR                   ;RX FAILED TO INTERRUPT
013010  042777  000100  166366  AABB1   BIC     #BIT6,@RXCSR    ;DISABLE RX INTERRUPTS
013016  104012                          SCOPE                   ;SCOPE
013020  012777  013040  166366  AABC1   MOV     #AABE,@RXVTR    ;HERE IF INTERRUPT OCCURS, CHANGE SERVICE TO
013026  012716  013034                  MOV     #AABD,@X6       ;AABE, SET EXIT POINTER TO AABD
013032  000002                          RTI                     ;EXIT INTERRUPT SERVICE
013034  000240                  AABD1   NOP                     ;OK IF NO INTERRUPT REOCCURS
013036  000764                          BR      AABB
013040  022626                  AABE1   POPSP2                  ;HERE IF INTERRUPT REOCCURS
013042  104003                          ERROR                   ;RX REINTERRUPTED AFTER RTI
013044  000761                          BR      AABB
                                        TSTA    100,,ABB,CD
                                        TSTAA   ABB,100,,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************************
013046  100053                  AT531   100053                  ;TEST NUMBER                                *
```

```
013050  013106                          AT54                    ;ADDRESS OF NEXT TEST                   *
013052  000144                          100.                    ;ITERATION COUNT                        *
013054  013056                          ABBA                    ;SCOPE ENTRY POINT                      *
        000053                          X=X+1                   ;                                       *
                                ;*****************************************************************
                                ;TEST THAT READING RXCSR DOES NOT CLEAR DONE BIT (RXCSR BIT 7 )
013056  004767  000304          ABBA:   JSR     %7,STRXD        ;SET RX DONE BIT
013062  017767  166316  166512          MOV     @RXCSR,RXCSRT   ;SAVE CONTENT OF RXCSR
013070  105777  166310                  TSTB    @RXCSR          ;SEE IF DONE BIT IS CLEAR
013074  100401                          BMI     ABBB            ;BRANCH IF DONE BIT IS NOT CLEAR
013076  104003                          ERROR
013100  005777  166302          ABBB:   TST     @RXBUF          ;CLEAR DONE BIT IF SET
013104  104012                          SCOPE                   ;SCOPE
                                        TSTA    100,,ACB,CD
                                        TSTAA   ACB,100,,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************
013106  100054                  AT54:   100054                  ;TEST NUMBER                            *
013110  013172                          AT55                    ;ADDRESS OF NEXT TEST                   *
013112  000144                          100.                    ;ITERATION COUNT                        *
013114  013122                          ACBA                    ;SCOPE ENTRY POINT                      *
        000054                          X=X+1                   ;                                       *
                                ;*****************************************************************
                                ;TEST THAT DONE CAN CAUSE INT WITH ERROR SET
013116  104006                          STRXV                   ;SET RX INTERRUPT SERVICE TO ACBB,
013120  013160                          ACBB
013122  004767  000240          ACBA:   JSR     %7,STRXD        ;SET RX DONE BIT
013126  004767  000234                  JSR     %7,STRXD        ;SET RX DATA OFLOW
013132  042777  000100  166244          BIC     #BIT6,@RXCSR    ;DISABLE RX INTERRUPTS
013140  005067  164632                  CLR     PSW             ;SET PROCESSOR PRIORITY TO 0
013144  052777  000100  166232          BIS     #BIT6,@RXCSR    ;ENABLE RX INTERRUPTS
013152  000240                          NOP
013154  104003                          ERROR                   ;RX DONE FAILED TO CAUSE INTERRUPT
013156  000401                          BR      ACBC
013160  022626          ACBB:   POPSP2                          ;HERE IF INTERRUPT OCCURS, POP STOCK TWICE
013162  042777  000100  166214  ACBC:   BIC     #BIT6,@RXCSR
013170  104012                          SCOPE
                                        TSTA    3,,ADD,CD
                                        TSTAA   ADD,3,,\X+1+CD,\X+2,\X+1
                                ;*****************************************************************
013172  100055                  AT55:   100055                  ;TEST NUMBER                            *
013174  013214                          AT56                    ;ADDRESS OF NEXT TEST                   *
013176  000003                          3.                      ;ITERATION COUNT                        *
013200  013206                          ADDA                    ;SCOPE ENTRY POINT                      *
        000055                          X=X+1                   ;                                       *
                                ;*****************************************************************
                                ;DATA TEST USING NORMAL CONFIGURATION
013202  004767  167222                  JSR     %7,CDINIT       ;INIT IF C=D DEVICE
013206  004567  171126          ADDA:   JSR     5,DATTST
013212  104012                          SCOPE
                                        TSTA    3,,APB,0
                                        TSTAA   APB,3,,\X+1+0,\X+2,\X+1
                                ;*****************************************************************
013214  000056                  AT56:   56                      ;TEST NUMBER                            *
013216  013306                          AT57                    ;ADDRESS OF NEXT TEST                   *
013220  000003                          3.                      ;ITERATION COUNT                        *
```

```
013222  013230                          APBA                    ;SCOPE ENTRY POINT                      *
        000056                          X=X+1                   ;                                       *
                                ;*******************************************************************
                                ;DATA TEST USING JUMPER CONNECTOR,
                                ;USES SPECIAL BINARY COUNT PATTERN FOR DATA, NO INTERRUPT,
013224  004767  170442                  JSR     7,INBIN         ;INITIALIZE BINARY COUNT PATTERN
013230  012767  001750  166322  APBA1:  MOV     #1000.,CTR0     ;SET CHARACTER COUNT TO 1000
013236  104020                  APBB1:  TIMETX                  ;TIME OUT TX DONE
013240  104003                          ERROR                   ;ERROR DONE NOT SETTING
013242  004767  170530                  JSR     7,GTBINP        ;GET BINARY CHARACTER
013246  110167  166302                  MOVB    %1,CRBUFA       ;SAVE CHAR IN CRBUFA AND
013252  004767  171466                  JSR     7,MASKIT        ;MASK OFF NON TRANSMITTED BITS
013256  110177  166130                  MOVB    %1,@TXBUF       ;LOAD CHAR,
013262  104017                          TIMERX                  ;TIME OUT RX DONE
013264  104003                          ERROR                   ;ERROR DONE NOT SETTING
013266  117767  166114  166256          MOVB    @RXBUF,CRBUF    ;LOAD RECEIVED DATA INTO CRBUF
013274  104004                          DATCHK                  ;CHECK DATA
013276  005367  166256                  DEC     CTR0            ;TESTED 1000 CHARACTERS
013302  001355                          BNE     APBB            ;BRANCH IF NOT
013304  104012                          SCOPE                   ;YES, SCOPE
                                        TSTAA   AQB,10.,,\X+1+CD,LAST,\X+1
                                ;*******************************************************************
013306  100057                  AT571:  100057                  ;TEST NUMBER                            *
013310  177777                          ATLAST                  ;ADDRESS OF NEXT TEST                   *
013312  000012                          10.                     ;ITERATION COUNT                        *
013314  013316                          AQBA                    ;SCOPE ENTRY POINT                      *
        000057                          X=X+1                   ;                                       *
                                ;*******************************************************************
                                ;TEST THAT WHEN TXCSR BIT 0 IS SET THAT THE OUTPUT DATA LINE
                                ;IS PULLED TO A SPACE,
013316  004767  167106          AQBA1:  JSR     %7,CDINIT       ;INIT IF C=D DEVICE
013322  052777  000004  166060          BIS     #BIT2,@TXCSR    ;SET MAINTENANCE BIT IN TXCSR
013330  052777  000001  166052          BIS     #BIT0,@TXCSR    ;SET BREAK BIT
013336  012777  000252  166046          MOV     #252,@TXBUF     ;LOAD BUFFER
013344  104017                          TIMERX                  ;TIME OUT RX DONE
013346  104003                          ERROR                   ;ERROR DONE NOT SETTING
013350  127727  166032  000000          CMPB    @RXBUF,#0       ;CHARACTER RECEIVED SHOULD BE 0
013356  001401                          BEQ     .+4
013360  104003                          ERROR                   ;CHARACTER OTHER THAN 0
013362  104011                          SRESET                  ;ISSUE RESET
013364  104012                          SCOPE
```

```
                                        ;
                                        ;SUBROUTINE TO SET RXCSR DONE BIT,
013366  052777  000004  166014  STRXD:  BIS     #BIT2,@TXCSR    ;SET MAINTENANCE BIT,
013374  005077  166012                  CLR     @TXBUF          ;LOAD TXBUF,
013400  104017                          TIMERX                  ;TIME OUT TX DONE
013402  104003                          ERROR                   ;ERROR DONE NOT SETTING
013404  000207                          RTS     %7              ;EXIT,
                                        ;SUBROUTINE TO CHECK THAT CTR0 THROUGH CTR3 CONTAIN DESCENDING VALUES,
013406  026767  166146  166146  CMPT:   CMP     CTR0,CTR1
013414  101430                          BLOS    CMPTNG
013416  026767  166140  166140          CMP     CTR1,CTR2
013424  101424                          BLOS    CMPTNG
013426  026767  166132  166132          CMP     CTR2,CTR3
013434  101420                          BLOS    CMPTNG
013436  026767  166124  166124          CMP     CTR3,CTR4
013444  101414                          BLOS    CMPTNG
013446  026767  166116  166116          CMP     CTR4,CTR5
013454  101410                          BLOS    CMPTNG
013456  026767  166110  166110          CMP     CTR5,CTR6
013464  101404                          BLOS    CMPTNG
013466  026767  166102  166102          CMP     CTR6,CTR7
013474  101002                          BHI     CMPTOK
013476  062716  000002          CMPTNG: ADD     #2,@X6
013502  000207                  CMPTOK: RTS     %7
```

```
                                        ;**********
                                        ;PRG1 - TRANSMITTER SCOPE LOOP
                                        ;**********
013504  104000                  PRG1:   TYPE                            ;TYPE PROGRAM TITLE.
013506  014541                          .RAD50
013510  004567  167766                  JSR     5,LINSLX                ;GO GET LINE # FROM USER
013514  104000                          TYPE                            ;TYPE SELECT CHAR AND DELAY.
013516  014636                          .SELCAD
013520  000000                          HALT                            ;WAIT FOR USER.
013522  116767  164042  000010  PRG1A: MOVB    SR,PRG1B                ;DELAY COUNT TO PRG1B.
013530  116777  164035  165654          MOVB    SR+1,@TXBUF             ;LOAD TXBUF.
013536  104016                          DELAY                           ;DELAY # OF MSECS, SET AT SR.
013540  000000                  PRG1B:  .BYTE
013542  000767                          BR      PRG1A                   ;REPEAT.
                                        ;**********
                                        ;PRG2 - RECEIVER SCOPE LOOP.
                                        ;**********
013544  104000                  PRG2:   TYPE                            ;TYPE PROGRAM TITLE.
013546  014601                          .RAD50
013550  004567  167726                  JSR     5,LINSLX                ;GO GET LINE # FROM USER
013554  104000                          TYPE                            ;TYPE SELECT CHAR AND DELAY.
013556  014636                          .SELCAD
013560  000000                          HALT                            ;WAIT FOR USER.
013562  052777  000004  165620  PRG2A: BIS     #BIT2,@TXCSR            ;SET MAINTENANCE BIT.
013570  116767  163774  000010          MOVB    SR,PRG2B                ;DELAY COUNT TO PRG2B.
013576  116777  163767  165606          MOVB    SR+1,@TXBUF             ;LOAD TXBUF.
013604  104016                          DELAY                           ;DELAY # OF MSECS, SET IN SR.
013606  000000                  PRG2B:  .BYTE
013610  017700  165572                  MOV     @RXBUF,R0               ;RXBUF CONTENTS TO R0.
013614  000005                          RESET                           ;DISPLAY CONTENTS OF RXBUF (IN R0).
013616  000005                          RESET                           ;BY ISSUING 5 RESET INSTRUCTIONS
013620  000005                          RESET
013622  000005                          RESET
013624  000005                          RESET
013626  000755                          BR      PRG2A
```

```
                                        ;**********
                                        ;PRG3 - SINGLE CHARACTER MAINTENANCE MODE DATA TEST;
                                        ;**********
013630  104020                  PRG3:   TYPE                    ;TYPE PROGRAM TITLE.
013632  015355                          P3TIT
013634  004567  167642                 JSR     5,LINSLX        ;GO GET LINE # FROM USER
013640  104000                          TYPE                    ;TYPE| SELECT CHARACTER;
013642  015500                          SELCAR
013644  000000                          HALT
013646  116767  163717  165700 PRG3A:  MOVB    SR+1,CRBUFA     ;MOVE DATA CHAR TO C 'UFA;
013654  004767  000034                 JSR     %7,MOUTIN       ;GO OUTPUT, RECEIVE, A'D CHECK DATA,
013660  000772                         BR      PRG3A
                                        ;**********
                                        ;PRG4 - SPECIAL BINARY COUNT MAINTENANCE MODE DATA TEST;
                                        ;**********
013662  104000                  PRG4:   TYPE                    ;TYPE PROGRAM TITLE;
013664  015425                          P4TIT
013666  004567  167610                 JSR     5,LINSLX        ;GO GET LINE # FROM USER
013672  004767  167774                 JSR     %7,INBIN        ;INITIALIZE BINARY COUNT;
013676  004767  170074         PRG4A:  JSR     %7,GTBINP       ;GET BINARY CHARACTER;
013702  110167  165646                 MOVB    %1,CRBUFA       ;SAVE AT CRBUFA;
013706  004767  000002                 JSR     %7,MOUTIN       ;GO OUTPUT, RECEIVE, AND CHECK DATA,
013712  000771                         BR      PRG4A           ;REPEAT;
                                        ;SUBROUTINE TO OUTPUT, RECEIVE, AND CHECK DATA WITH MAINTENANCE BIT SET,
013714  032767  000200  163646 MOUTIN: BIT     #BIT7,SR        ;SEE IF BIT 7 IS SET;
013722  001001                         BNE     .+4             ;BRANCH IF SET;
013724  104002                         STALL                   ;SET, DO A RANDOM STALL;
013726  104020                         TIMETX                  ;TIME OUT TX DONE
013730  104003                         ERROR                   ;ERROR DONE NOT SETTING
013732  052777  000004  165450         BIS     #BIT2,@TXCSR    ;SET MAINTENANCE BIT;
013740  016777  165610  165444         MOV     CRBUFA,@TXBUF   ;LOAD TXBUF,
013746  004767  170772                 JSR     7,MASKIT        ;MASK OFF NON TRANSMITTED BITS
013752  104017                         TIMERX                  ;TIME OUT RX DONE
013754  104003                         ERROR                   ;ERROR DONE NOT SETTING
013756  017767  165424  165566         MOV     @RXBUF,CRBUF    ;MOVE CHAR IN RX BUFFER TO CRBUF;
013764  104004                         DATCHK                  ;COMPARE EXPECTED AND RECEIVED DATA
013766  000207                         RTS     %7              ;EXIT;
```

```
                                        ;ASCII MESSAGES
                                        ;
013770  042045  030514  026461  MTITI   ,ASCII  '%DL11-E,C/D OFF LINE TEST - MAINDEC-11-DEDLA-A%'
013776  026105  027523  020104
014004  043117  020106  044514
014012  042516  052040  051505
014020  020124  020055  040515
014026  047111  042504  026503
014034  030461  042055  042132
014042  040514  040455     045
014047     045  040515  020120          ,ASCII  '%MAP OF DEVICES PRESENT%@'
014054  043117  042040  053105
014062  041511  051505  050040
014070  042522  042523  052116
014076  040045
014100  020040  020040  020040  MDEVADI ,ASCII  '               %@'
014106  020040  020040  020040
014114  020040  040045
014120  047045  047117  020105  MNONEI  ,ASCII  '%NONE FOUND%@'
014126  047506  047125  022504
014134     100
014135     045     124          EM0I    ,ASCII  '%T'
014137     040  020040  050040  ATNUMBI ,ASCII  '    PC= '
014144  036503     040
014147     040  020040  020040  APCI    ,ASCII  '        RXCSR= '
014154  020040  051040  041530
014162  051123  020075
014166  020040  020040  020040  MRXNUMI ,ASCII  '        @'
014174  020040     100
014177     045  050045  043522  P0TITI  ,ASCII  '%%PRG0 - INPUT-OUTPUT LOGIC TESTS' '
014204  020060  020055  047111
014212  052520  026524  052517
014220  050124  052125  046040
014226  043517  041511  052040
014234  051505  051524  020056
014242  042045  051511  047503          ,ASCII  '%DISCONNECT DL11-E FROM MODEM'
014250  047116  041505  020124
014256  046104  030461  042455
014264  043040  047522  020115
014272  047515  042504     115
014277     040  047101  020104          ,ASCII  ' AND CONNECT JUMPER TO CABLE,X@'
014304  047503  047116  041505
014312  020124  052512  050115
014320  051105  052040  020117
014326  040503  046102  027105
014334  040045
014336  054124  051503  020122  ATXCSRI ,ASCII  'TXCSR S/B= '
014344  027523  035102     040
014351     040  020040  020040  ATXSBI  ,ASCII  '        WAS= '
014356  020040  053440  051501
014364  020072
014366  020040  020040  020040  ATXWASI ,ASCII  '       @'
014374     100
014375     122  041530  051123  ARXCSRI ,ASCII  'RXCSR S/B= '
```

```
014472  051440  041057  020072
014410  020040  020040  020040  ARXSBI  ,ASCII  /        WASI /
014416  020040  040527  035123
014424     040
014425     040  020040  020040  ARXWASI ,ASCII  /        @/
014432  020040     100
014435     124  020130  050123  ETXTIMI ,ASCII  /TX SPEEDS NOT IN ASCENDING ORDER;@/
014442  042505  051504  047040
014450  052117  044440  020116
014456  051501  042503  042116
014464  047111  020107  051117
014472  042504  027122     100
014477     122  020130  050123  ERXTIMI ,ASCII  /RX SPEEDS NOT IN ASCENDING ORDER;@/
014504  042505  051504  047040
014512  052117  044440  020116
014520  051501  042503  042116
014526  047111  020107  051117
014534  042504  027122     100
014541     045  050045  043522  P1TITI  ,ASCII  /%%PRG1 - TRANSMITTER SCOPE LOOP@/
014546  020061  020055  051124
014554  047101  046523  052111
014562  042524  020122  041523
014570  050117  020105  047514
014576  050117     100
014601     045  050045  043522  P2TITI  ,ASCII  /%%PRG2 - RECEIVER SCOPE LOOP@/
014606  020062  020055  042522
014614  042503  053111  051105
014622  051440  047503  042520
014630  046040  047517  040120
014636  051445  052105  052040  SELCADI ,ASCII  /%SET TEST CHAR CODE IN SR15-SR8, SET DELAY TIME IN SR7-SR0,@/
014644  051505  020124  044103
014652  051101  041440  042117
014660  020105  047111  051440
014666  030522  026465  051123
014674  026070  051440  052105
014702  042040  046105  054501
014710  052040  046511  020105
014716  047111  051440  033522
014724  051455  030122  040056
014732  040504  040524  051440  ERDATI  ,ASCII  /DATA S/R! /
014740  041057  020072
014744  020040  020040  040527  AASBI   ,ASCII  /    WASI /
014752  035123     040
014755     040  020040          AWASI   ,ASCII  /   /
014760  051040  041130  043125          ,ASCII  / RXBUF! /
014766  020072
014770  020040  020040  020040  ARXBUFI ,ASCII  /        @/
014776  040040
015000  051445  052105  042040  ASETSRI ,ASCII  /%SET DESIRED SR OPTIONS, NORMAL OPERATION /
015006  051505  051111  042105
015014  051440  020122  050117
015022  044524  047117  027123
015030  047040  051117  040515
015036  020114  050117  051105
015044  052121  047511  020116
```

```
015052  051511  053440  052111          ,ASCII  /IS WITH SR = 0000P0@/
015060  020110  051123  036440
015066  030040  030060  030060
015074  040060
015076  044445  041516  051117  AINCRT: ,ASCII  /%INCORRECT ROUTINE SELECTED, PLACE CORRECT PROGRAM/
015104  042522  052103  051040
015112  052517  044524  042516
015120  051440  046105  041505
015126  042524  026104  050040
015134  040514  042503  041440
015142  051117  042522  052103
015150  050040  047522  051107
015156  046501
015160  044445  020116  051123          ,ASCII  /%IN SR 0=2 AND PRESS CONTINUE',@/
015166  030040  031055  040440
015174  042116  050040  042522
015202  051523  041440  047117
015210  044524  052516  027105
015216     100
015217     045  047111  040526  AINCPG: ,ASCII  /%INVALID PROGRAM SELECTED',@/
015224  044514  020104  051120
015232  043517  040522  020115
015240  042523  042514  052103
015246  042105  040056
015252     207                  APGEND: ,BYTE   207
015253     045  047105  020104          ,ASCII  /%END PASS = /
015260  040520  051523  036440
015266     040
015267     040  020040  020040  APCNT:  ,ASCII  /        LINE =  /
015274  020040  046040  047111
015302  020105  020075     040
015307     040  020040  051040  ACLIN:  ,ASCII  /    RXCSR = /
015314  041530  051123  036440
015322     040
015323     040  020040  020040  APRXC:  ,ASCII  /        VECTOR = /
015330  020040  053040  041505
015336  047524  020122  020075
015344  020040  020040  020040  APVEC:  ,ASCII  /        @/
015352  020040     100
015355     045  050045  043522  P3TIT:  ,ASCII  /%%PRG3=SINGLE CHAR MAINT MODE DATA TEST@/
015362  026463  044523  043516
015370  042514  041440  040510
015376  020122  040515  047111
015404  020124  047515  042504
015412  042040  052101  020101
015420  042524  052123     100
015425     045  050045  043522  P4TIT:  ,ASCII  /%%PRG4=SPEC BIN COUNT MAINT MODE DATA TEST@/
015432  026464  050123  041505
015440  041040  047111  041440
015446  052517  052116  046440
015454  044501  052116  046440
015462  042117  020105  040504
015470  040524  052040  051505
015476  040124
015500  051445  052105  052040  SELCAR: ,ASCII  /%SET TEST CHAR CODE IN SR15=SR8,@/
```

```
015526  051505  020124  044103
015514  051131  041440  042117
015522  020105  047111  051440
015530  030522  026465  051123
015536  027070     100
015541     045  047514  042101  LDLINE: .ASCII  /%LOAD LINE NO, INTO SR #-2#/
015546  046040  047111  020105
015554  047516  020056  047111
015562  047524  051440  020122
015570  026460  040062
015574  020040  044514  042516  ALINE:  .ASCII  /  LINE NO,/
015602  047040  027117
015606  020040  053440  051501  SELINE: .ASCII  /   WAS SELECTED#/
015614  051440  046105  041505
015622  042524  040104
015626  051045  041505  044505  MSETRX: .ASCII  /%RECEIVER SPEED CHECK#/
015634  042526  020122  050123
015642  042505  020104  044103
015650  041505  040113
015654  052045  040522  051516  MSETTX: .ASCII  /%TRANSMIT SPEED CHECK#/
015662  044515  020124  050123
015670  042505  020104  044103
015676  041505  040113
015702  051445  052105  041440  MSETCL: .ASCII  /%SET CLOCK SWITCHES TO POSITION, THEN PRESS CONTINUE,#/
015710  047514  045503  051440
015716  044527  041524  042510
015724  020123  047524  050040
015732  051517  052111  047511
015740  026116  052040  042510
015746  020116  051120  051505
015754  020123  047503  052116
015762  047111  042525  040056
015770  042445  051122  051117  MTERR:  .ASCII  /%ERROR - UNEXPECTED TRAP/
015776  026440  052440  042516
016004  050130  041505  042524
016012  020104  051124  050101
016020  052045  040522  050120          .ASCII  /%TRAPPED TO  /
016026  042105  052040  020117
016034     040
016035     040  020040  020040  MTO:    .ASCII  /        /
016042  020040     040
016045     045  051124  050101          .ASCII  /%TRAPPED FROM PC  /
016052  042520  020104  051126
016060  046517  050040  020103
016066     040
016067     040  020040  020040  MFROM:  .ASCII  /        #/
016074  020040  040040
016100  047045  020117  042504  MNOLIN: .ASCII  /%NO DEVICE PRESENT - THIS LINE NO,#/
016106  044526  042503  050040
016114  042522  042523  052116
016122  026440  052040  044510
016130  020123  044514  042516
016136  047040  027117     100
016143     045  051503  036440  MS0:    .ASCII  /%[redacted] 0#/
016150  030040     100
```

```
016153      045  051503  036440  MS1:    ,ASCII  /%CS = 1@/
016160  030440      100
016163      045  051503  036440  MS2:    ,ASCII  /%CS = 2@/
016170  031040      100
016173      045  051503  036440  MS3:    ,ASCII  /%CS = 3@/
016200  031440      100
016203      045  051503  036440  MS4:    ,ASCII  /%CS = 4@/
016210  032040      100
016213      045  051503  036440  MS5:    ,ASCII  /%CS = 5@/
016220  032440      100
016223      045  051503  036440  MS6:    ,ASCII  /%CS = 6@/
016230  033040      100
016233      045  051503  036440  MS7:    ,ASCII  /%CS = 7@/
016240  033440      100
016243      045  042522  047503  MPWRF:  ,ASCII  /%RECOVERED FROM POWER FAILURE@/
016250  042526  042522  020104
016256  051106  046517  050040
016264  053517  051105  043040
016272  044501  052514  042522
016300      100
        016302                  ,EVEN
        000001                  ,END
```

```
AAA     006052      AAAA    007402      AAAB    007416      AAB     006764
AABA    012764      AABB    013010      AABC    013020      AABD    013034
AABE    013040      AAE     006766      AASB    014744      ABA     006104
ABAA    010334      ABAB    010426      ABB     006116      ABBA    013056
ABBB    013100      ABE     006120      ACA     006136      ACAA    010440
ACAB    010570      ACB     006150      ACBA    013122      ACBB    013160
ACBC    013162      ACE     006152      ACLIN   015307      ADA     006170
ADAA    010610      ADAB    010636      ADAC    010656      ADB     006202
ADDA    013206      ADE     006204      ADTENP  004320      AEA     006222
AEB     006236      AEC     006260      AED     006300      AFBA    007430
AFBB    007452      AFBC    007472      AGA     006322      AGB     006336
AGBA    007504      AGBB    007534      AGBC    007564      AGBD    007600
AGBE    007630      AGC     006360      AGD     006400      AIAA    010072
AIAF    011116      AIAS    011120      AIASA   011150      AIAST   011202
AINCPC  015217      AINCRT  015076      AJA     006422      AJB     006444
AJBA    007642      AJBB    007664      AJBC    007706      AJBD    007726
AJC     006466      AJD     006506      AKA     006530      AKB     006542
AKBA    007740      AKBC    007766      ALA     006554      ALAA    011214
ALAB    011244      ALAC    011256      ALBA    010116      ALBB    010140
ALBC    010162      ALBD    010202      ALINE   015574      ALY     006604
ALZ     006624      AMAA    011270      AMAC    011322      AMBA    010214
AMBB    010244      AMBC    010260      AMBD    010302      AMBE    010322
AOAA    011340      AOAB    011356      AOAB1   011372      AOAC    011414
AOAD    011426      AOBA    010020      AOBB    010042      AOBC    010064
AOBD    010104      APA     006636      APB     006652      APBA    013230
APBB    013236      APC     014147      APCNT   015267      APCX    006674
APD     006714      APGEND  015252      APRXC   015323      APVEC   015344
AQA     006736      AQAA    011444      AQAB    011670      AQAS    011672
AQASA   011714      AQAST   011734      AQB     006752      AQBA    013316
AQC     006774      AQD     007014      ARA     007036      ARAA    011746
ARAS    012062      ARB     007060      ARC     007102      ARD     007122
ARXBUF  014770      ARXCSR  014375      ARXSB   014410      ARXWAS  014425
ASA     007144      ASB     007166      ASC     007210      ASD     007230
ASETSR  015700      ATA     007252      ATAA    012122      ATAB    012154
ATAC    012156      ATB     007266      ATLAST = 177777     ATNUMB  014137
ATXCSR  014336      ATXSB   014351      ATXWAS  014366      AT0     006742
AT1     006774      AT10    006544      AT11    006626      AT12    006726
AT13    007026      AT14    007134      AT15    007242      AT16    007270
AT17    007316      AT2     006126      AT20    007344      AT21    007372
AT22    007420      AT23    007474      AT24    007632      AT25    007730
AT26    010010      AT27    010106      AT3     006160      AT30    010204
AT31    010324      AT32    010430      AT33    010600      AT34    010662
AT35    011204      AT36    011260      AT37    011324      AT4     006712
AT40    011434      AT41    011736      AT42    012102      AT43    012170
AT44    012246      AT45    012332      AT46    012430      AT47    012514
AT5     006312      AT50    012576      AT51    012660      AT52    012750
AT53    013046      AT54    013106      AT55    013172      AT56    013214
AT57    013306      AT6     006412      AT7     006520      AUAA    012204
AUAB    012230      AUAC    012240      AVAA    012262      AVAB    012320
AVAC    012322      AWAA    012342      AWAB    012372      AWAC    012402
AWAD    012416      AWAE    012422      AWAS    014755      AXA     007300
AXAA    012454      AXAB    012502      AXAC    012504      AXB     007314
AX1A    012534      AX1B    012564      AX1C    012566      AYA     007326
AYAA    012616      AYAB    012642      AYAC    012642      AYB     007342
AZA     007354      AZAA    012700      AZAB    012736      AZAC    012748
```

```
A&B       007370          BDCNV     004154          BDCNVA    004206          BDCNVB    004240
BDCNVC    004242          BDCNVD    004244          BIT0   =  000001          BIT1   =  000002
BIT10  =  002000          BIT11  =  004000          BIT12  =  010000          BIT13  =  020000
BIT14  =  040000          BIT15  =  100000          BIT2   =  000004          BIT3   =  000010
BIT4   =  000020          BIT5   =  000040          BIT6   =  000100          BIT7   =  000200
BIT8   =  000400          BIT9   =  001000          BMOVA     004142          BMOVB     004132
CD     =  100000          CDINIT    002430          CDINX     002454          CHAINN    002322
CHALT  =  104005          CHNA      002340          CHNAA     002336          CHNAB     002332
CHNB      002402          CLRCD     003022          CMAS0     001300          CMAS1     001302
CMAS10    001320          CMAS11    001322          CMAS12    001324          CMAS13    001326
CMAS14    001330          CMAS15    001332          CMAS16    001334          CMAS17    001336
CMAS2     001304          CMAS20    001340          CMAS21    001342          CMAS22    001344
CMAS23    001346          CMAS24    001350          CMAS25    001352          CMAS26    001354
CMAS27    001356          CMAS3     001306          CMAS30    001360          CMAS31    001362
CMAS32    001364          CMAS33    001366          CMAS34    001370          CMAS35    001372
CMAS36    001374          CMAS4     001310          CMAS5     001312          CMASK     001314
CMAS7     001316          CMPT      013406          CMPTNG    013476          CMPTOK    013502
CNVCTR    004312          CONT      004672          CRBUF     001552          CRBUFA    001554
CRBUFB    001556          CTR0      001560          CTR1      001562          CTR2      001564
CTR3      001566          CTR4      001570          CTR5      001572          CTR6      001574
CTR7      001576          CURTST    001450          DATAA     004362          DATCHK =  104004
DATTST    004340          DECVAL    004332          DELAY  =  104016          DIGIT     004314
DLCNT     003342          DLY       003276          DLYA      003320          DLYB      003324
DLYC      003336          DOTHIS    005566          DTCHK     004774          DTCHKA    005064
DTCHKX    005022          EHALT  =  104010          EHLT      004732          EHLTA     004742
EMTA      002522          EMTINT    002510          EMTTAB    001510          EMT       014135
ERDAT     014732          ERR       005066          ERRA      005136          ERRB      005236
ERRC      005242          ERRD      005252          ERRE      005254          ERROR  =  104003
ERROR1 =  104015          ERRL      005110          ERTP      005256          ERTPA     005326
ERXTIM    014477          ETXTIM    014435          FMAP      001466          FNONE     001620
FORMAD    005442          FORWD     002456          FOUNDV    001606          FROMPC    001624
FTITLE    001616          GETRDY    002164          GOBACK    005602          GRCNT     003370
GTBIN     003730          GTBINP    003776          GTRDA1    002254          GTRDYA    002224
GTRDYC    002262          GTRDYD    002304          GTRDYX    002172          ICTR      001460
INBIN     003672          INCRPG    004460          INCRTN    004450          KSTART    001446
LDLINE    015541          LINA      003564          LINB      003572          LINEND    001610
LINSEL    003464          LINSLX    003502          LOGIC     004662          MACHER    000004
MANUAL =  100000          MAPA      001722          MAPEND    002010          MAPERR    002074
MAPNE     001750          MAPOK     001760          MAPVEC    005364          MASKIT    004744
MDEVAD    014100          MEROM     016067          MNOLIN    016100          MNONE     014120
MOUTIN    013714          MPWRF     016243          MRXNUM    014166          MSETC     015702
MSETRX    015626          MSETTX    015654          MS0       016143          MS1       016153
MS2       016163          MS3       016173          MS4       016203          MS5       016213
MS6       016223          MS7       016233          MTERR     015770          MTIT      013770
MTO       016035          NOP    =  000240          NXTST     001456          OACNV     004044
OACNVA    004060          OACNVX    004130          OLDPS     001464          OPEN   =  000000
OVRLAY    003240          OVRLYA    003254          OVRLYB    003274          PASCNT    001626
PFAIL     005604          PIND      003722          POPSP  =  005726          POPSP2 =  022626
PRGEA     004620          PRGER     004704          PRGEC     004612          PRGEND    004472
PRGEXT    004562          PRGNUM    001444          PRGTAB    001470          PRGXTL    004652
PRGXT1    004600          PRG0      006004          PRG0A     006036          PRG0B     006032
PRG1      013504          PRG1A     013522          PRG1B     013540          PRG2      013544
PRG2A     013562          PRG2B     013606          PRG3      013630          PRG3A     013646
PRG4      013662          PRG4A     013676          PRTY0  =  000000          PRTY1  =  000040
PRTY2  =  000100          PRTY3  =  000140          PRTY4  =  000200          PRTY5  =  000240
```

```
PRTY6  = 000300        PRTY7  = 000340        PSW    = 177776        PT2    003716
PT0P     003724        PT1      003720        PT1P     003726        PWRUP  000014
P0TIT    014177        P1TIT    014341        P2TIT    014401        P3TIT  015155
P4TIT    015425        RCMSK    003410        RENAP    001630        RESTAR 005722
RIND     003714        RMASK    001400        RNCNT    003412        RNSEN  002750
RP1      003716        RP2      003020        RSTART   005764        RSTPC  002636
RSTPSW   002640        RSTREG = 104014        RSTRG    002602        RTNNO  001452
RXBUF    001406        RXBUFT   001604        RXCR0    001200        RXCR1  001202
RXCR10   001220        RXCR11   001222        RXCR12   001224        RXCR13 001226
RXCR14   001230        RXCR15   001232        RXCR16   001234        RXCR17 001236
RXCR2    001204        RXCR20   001240        RXCR21   001242        RXCR22 001244
RXCR23   001246        RXCR24   001250        RXCR25   001252        RXCR26 001254
RXCR27   001256        RXCR3    001206        RXCR30   001260        RXCR31 001262
RXCR32   001264        RXCR33   001266        RXCR34   001270        RXCR35 001272
RXCR36   001274        RXCR4    001210        RXCR5    001212        RXCR6  001214
RXCR7    001216        RXCSR    001404        RXCSRT   001602        RXEND  001276
RXLVL    001416        RXVTR    001414        SAVREG = 104013        SAVRG  002542
SCOPE  = 104012        SCOPTR   001462        SELCAD   014636        SELCAR 015500
SELINE   015606        SETSR    004440        SIOT     003462        SPROT  = 001176
SR     = 177570        SRESET = 104011        SRSETT   002732        STAL   003344
STALA    003364        STALB    003366        STALL  = 104002        START  001634
START1   002024        START2   002030        START3   002764        START4 002124
STLMSK   001402        STLSPV   002676        STLSRV   002642        STPPA  002720
STPRA    002664        STRXB    013366        STRXV  = 104006        STTXV  = 104007
SUBTEN   004252        SUBTNA   004256        SUBTNB   004272        SVPPC  002576
SVRPSW   002600        TEMP     001612        TEMP1    001614        TENPWR 004316
TIME     011170        TIMER    003460        TIMERX = 104017        TIMETX = 104020
TIMEX    003456        TIME1    003432        TIME2    003436        TIM1   011174
TKB      001426        TKLVL    001436        TKS      001424        TKVTR  001434
TMRX     003414        TMTX     003424        TNNO     001454        TOPC   001622
TPB      001432        TPLVL    001442        TPS      001430        TPVTR  001440
TSTVA    005730        TSTVEC   005640        TSTVEX   005750        TXBUF  001412
TXCSR    001410        TXCSRT   001600        TXLVL    001422        TXVTR  001420
TYP      003064        TYPA     003074        TYPC     003112        TYPD   003140
TYPDAT   003204        TYPE   = 104000        TYPES  = 104001        TYPF   003156
TYPG     003170        TYPS     003206        TYPSA    003232        TYPSR  003234
UMASK    001376        X      = 000057        .      = 016302
```

ERRORS DETECTED: 0